National Library of Canada
Bibliothèque nationale du Canada

Canadian Theses Service
Services des thèses canadiennes

Ottawa, Canada
K1A 0N4

CANADIAN THESES
THÈSES CANADIENNES

NOTICE

The quality of this microfiche is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Previously copyrighted materials (journal articles, published tests, etc.) are not filmed.

Reproduction in full or in part of this film is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30. Please read the authorization forms which accompany this thesis.

AVIS

La qualité de cette microfiche dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

Les documents qui font déjà l'objet d'un droit d'auteur (articles de revue, examens publiés, etc.) ne sont pas microfilmés.

La reproduction, même partielle, de ce microfilm est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30. Veuillez prendre connaissance des formules d'autorisation qui accompagnent cette thèse.

THIS DISSERTATION
HAS BEEN MICROFILMED
EXACTLY AS RECEIVED

LA THÈSE A ÉTÉ
MICROFILMÉE TELLE QUE
NOUS L'AVONS REÇUE

NL 339 (r. 86/01)

Canada

National Library of Canada
Bibliothèque nationale du Canada

104

Ottawa, Canada
K1A 0N4

TC –

0-315-23301-X

CANADIAN THESES ON MICROFICHE SERVICE – SERVICE DES THÈSES CANADIENNES SUR MICROFICHE

PERMISION TO MICROFILM – AUTORISATION DE MICROFILMER

• Please print or type – Écrire en lettres moulées ou dactylographier

AUTHOR – AUTEUR

Full Name of Author – Nom complet de l'auteur

LIM - CHO SIU

Date of Birth – Date de naissance	Canadian Citizen – Citoyen canadien
NOV. 19, 1957	☐ Yes Oui ☑ No Non
Country of Birth – Lieu de naissance	**Permanent Address – Résidence fixe**
HONG KONG	17A WAH SHAN MANSION TAI KOO SHING, QUARRY BAY HONG KONG

THESIS – THÈSE

Title of Thesis – Titre de la these

APPLICATION OF ADAPTIVE DIFFERENTIAL PULSE-CODE MODULATION IN FEEDBACK CONTROL SYSTEMS

Degree for which thesis was presented Grade pour lequel cette these fut présentee	Year this degree conferred Année d'obtention de ce grade
M.Sc	1985
University – Université	**Name of Supervisor – Nom du directeur de these**
UNIVERSITY OF ALBERTA	Dr. R.E. RINK

AUTHORIZATION – AUTORISATION

Permission is hereby granted to the NATIONAL LIBRARY OF CANADA to **microfilm** this thesis and to **lend** or **sell copies of the film.**

The author reserves other publication rights, and neither the thesis nor extensive extracts from it may be printed or otherwise reproduced without the author's written permission.

L'autorisation est, par la présente, accordée à la BIBLIOTHÈQUE NATIONALE DU CANADA de **microfilmer** cette thèse et de **prêter** ou de **vendre des exemplaires du film.**

L'auteur se réserve les autres droits de publication; ni la thèse ni de longs extraits de celle-ci ne doivent être imprimés ou autrement reproduits sans l'autorisation écrite de l'auteur.

ATTACH FORM TO THESIS – VEUILLEZ JOINDRE CE FORMULAIRE À LA THÈSE

Signature	Date
Lim Cho Siu	May 17, 1985

NL-91 (r. 84/03)

Canada

THE UNIVERSITY OF ALBERTA

Application of Adaptive Differential Pulse-Code Modulation in Feedback Control Systems

by

Lim-Cho Siu

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES AND RESEARCH

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE

OF Master of Science

Department of Electrical Engineering

EDMONTON, ALBERTA

Fall, 1985

THE UNIVERSITY OF ALBERTA

RELEASE FORM

NAME OF AUTHOR Lim-Cho Siu

TITLE OF THESIS Application of Adaptive Differential Pulse-Code Modulation in Feedback Control Systems

DEGREE FOR WHICH THESIS WAS PRESENTED Master of Science

YEAR THIS DEGREE GRANTED Fall, 1985

Permission is hereby granted to THE UNIVERSITY OF ALBERTA LIBRARY to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly or scientific research purposes only.

The author reserves other publication rights, and neither the thesis nor extensive extracts from it may be printed or otherwise reproduced without the author's written permission.

(SIGNED) Lim Cho Siu.....

PERMANENT ADDRESS:

17A. WAH SHAN MANSION
TAI KOO SHING, QUARRY BAY
HONG KONG.....

DATED .....May 6.....1985

THE UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES AND RESEARCH

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research, for acceptance, a thesis entitled Application of Adaptive Differential Pulse-Code Modulation in Feedback Control Systems submitted by Lim-Cho Siu in partial fulfilment of the requirements for the degree of Master of Science.

R. E. Rink

Supervisor

M. M. Gupta

V. Gourishankar

Date 6 May 1985

ABSTRACT

It is well known that analog-to-digital (A/D) conversion error, commonly called quantization noise, can affect significantly the performance of a feedback control system. Furthermore, there is always a conflict between the quantization noise and the A/D dynamic range. The problem becomes more important when the A/D wordlength is rather limited. This problem has not been studied extensively in the control systems area but has received much attention in the communication systems area. Many interesting and useful techniques for efficient quantization and coding have resulted. One particular scheme, known as Adaptive Differential Pulse Code Modulation (ADPCM), is found to be very effective in extending the dynamic range while maintaining an acceptable level of quantization error, even with limited A/D wordlength.

This thesis is concerned with the application of an ADPCM coder as an A/D converter to digital feedback control systems. The main purpose of this work is to study the merits that can result from such an application. The research is oriented to practical hardware implementation considerations. Before the proposed system is presented, a brief review on the subject of differential encoding in communication is given. Then the configuration of the proposed system is presented and discussed. Design methods for the control system and the ADPCM coder are also explained and discussed. Suggestions for the implementation

of the control system with practical hardware components are also included and discussed briefly.

The performance of the control system is analysed theoretically by means of linear transfer function analysis, but with a few simplifying assumptions. The resulting performance estimate is only a crude approximation to the ADPCM system, and is simply that of a nonadaptive DPCM system. Furthermore, the issue of stability for the proposed control system is also presented very briefly. In order to evaluate the performance of the proposed system, computer simulations are performed with two specific examples, and performance results are compared with that of a fixed A/D system. Comparison of simulation results indicate that, in general, ADPCM yields superior performance to that of a fixed A/D converter with the same wordlength. The simulation results also agree reasonably well, in many cases, with the performance estimate obtained from the simplified approximate analysis. It is observed, however, that nonlinear effects such as limit cycles exist in some cases and this demonstrates the nonlinear nature of the ADPCM coder. Furthermore, the simulation results also demonstrate that instability may occur for open-loop unstable systems when there is inadequate feedback information. Finally, summary and conclusions of this research is given and suggestions for further study are also included.

ACKNOWLEDGEMENT

I would like to take this opportunity to express my sincere appreciation to Dr. R. E. Rink, thesis supervisor, teacher, and friend, for his continual advice, encouragement, support, understanding, patience, and kindness in the entire course of this research project. I would also like to thank Dr. V. Gourishankar and Dr. S. Shah for their interest in serving on the examining committee.

I am very grateful to the Department of Electrical Engineering for financial support of this project. It is also supported by the National Science and Engineering Research Council under Grant No. A-5629. The library resources and computing facilities at the University of Alberta must also be acknowledged. Without these, the task of carrying out the research project would not have been possible.

I would like to express my sincere gratitude to all my friends who have directly and indirectly provided me with warm encouragement and support throughout my research project. I am also grateful for their willingness to provide assistance whenever it is needed. In particular, I would like to thank Y. Ung for proof-reading part of the manuscript of this thesis.

Finally, I would like to give thanks to my parents, sisters, and brother for their incessant encouragement, understanding, and patience. Without their constant support, this thesis would not have been possible.

TABLE OF CONTENTS

Chapter — Page

1. INTRODUCTION 1

1.1 Background 1

1.2 Objective and Scope of Thesis 5

1.3 Outline of Thesis 6

2. ADAPTIVE DIFFERENTIAL QUANTIZATION IN DIGITAL WAVEFORM COMMUNICATION 8

2.1 Introduction 8

2.2 Differential Encoding System 12

2.2.1 Basic System Description 12

2.2.2 Signal-to-Quantization Noise Ratio Improvement 15

2.2.3 Predictor 17

2.2.4 Quantizer 20

2.3 Adaptive Differential Encoding System 23

3. ADAPTIVE DIFFERENTIAL QUANTIZATION IN FEEDBACK CONTROL 32

3.1 Introduction 32

3.2 System Configuration and Description 35

3.3 Controller 39

3.4 Predictor 43

3.5 Adaptive Quantizer 49

3.5.1 Feedback Adaptation 55

3.5.2 Feedforward Adaptation 59

3.5.3 Selection of Adaptation Threshold Parameter 60

3.5.4 Quantizer Scaling 63

4. SUGGESTED HARDWARE REALIZATIONS AND IMPLEMENTATION CONSIDERATIONS 71

4.1 Introduction 71

4.2 Adaptive Quantizer Realization 72

4.3 Adaptive Step-Size Control 76

4.3.1 Feedback Algorithm 79

4.3.2 Feedforward Algorithm 81

4.4 Internal Scaling of Data 84

4.4.1 Transfer Function Realizations 85

4.4.1.1 First-Order Transfer Function 86

4.4.1.2 Second-Order Transfer Function 89

4.4.2 ADPCM Prediction Feedback Loop Scaling 91

5. PERFORMANCE ANALYSIS AND SIMULATION RESULTS 95

5.1 Introduction 95

5.2 Performance Analysis 96

5.3 Computer Simulation Results 102

5.3.1 ADPCM in a Marginally Stable Plant 107

5.3.1.1 Performance Versus *C* 112

5.3.1.2 Performance Versus *M* 122

5.3.1.3 Performance Comparison of ADPCMQ and ADPCMF Coders 135

5.3.2 Performance Comparisons of ADPCM Coders and Fixed A/D Converters 139

5.3.2.1 Steady-State Tracking Performance 143

5.3.2.2 Transient Performance Comparisons 154

5.3.3 ADPCM in an Unstable Plant 165

5.3.3.1 Performance Versus *C* 169

5.3.3.2 Steady-State Performance Comparisons 172

5.3.3.3 Transient Performance Comparisons 176

5.4 Summary 178

6. CONCLUSIONS 180

6.1 Summary and Conclusions 180

6.2 Suggestions for Further Study 184

BIBLIOGRAPHY 188

APPENDIX A - Derivation of Steady-State LQG Compensator Transfer Function for a System with Non-Zero Set-Point 194

APPENDIX B - Derivation of Equations (3.40) and (3.41) ..198

APPENDIX C - Steady-State Optimal State Feedback Gain and Steady-State Kalman Filter Gain Design Data 202

APPENDIX D - Quantizer Scaling Design Data 204

LIST OF TABLES

Table	Description	Page
C.1	Steady-State Optimal State Feedback Gain and Closed-Loop Pole Locations for Plant A.	202
C.2	Steady-State Optimal State Feedback Gain and Closed-Loop Pole Locations for Plant B.	202
C.3	Steady-State Kalman Filter Gain and Closed-Loop Pole Location for Plant A.	203
C.4	Steady-State Kalman Filter Gain and Closed-Loop Pole Location for Plant B.	203
D.1	Adaptive Quantizer Scaling for Plant A.	204
D.2	Adaptive Quantizer Scaling for Plant B.	204
D.3	Quantizer Scaling of Fixed A/D for Plant A with 3σ Loading and $b=4$.	205

LIST OF FIGURES

Figure		Page
1.1	Input-Output Characteristic of a Uniform b-bit Mid-Riser Quantizer.	3
2.1	Differential Encoding Communication System.	13
2.2	ADPCM System with Feedback Adaptive Quantizer [After Rabiner and Schafer (1978)].	24
2.3	ADPCM System with Feedforward Adaptive Quantizer [After Rabiner and Schafer (1978)].	25
3.1	Adaptive Differential Quantization Scheme in a Control Loop [After Rink (1982)].	36
3.2	Input-Output Characteristic of a b-Bit Adaptive Quantizer.	50
4.1	Alternative Configuration of an Adaptive Quantizer [After Johnston and Goodman (1978)].	73
4.2	Realization of Adaptive Quantizer Pre-Scaling.	75
4.3	Realizaiton of Analog Output Re-Scaling.	76
4.4	Feedback Step-Size Adaptation Implementation.	80
4.5	Feedforward Step-Size Adaptation Implementation.	82
4.6	Realization of a First-Order Transfer Function with Consistent Internal Scaling.	88
4.7	Realization of a Second-Order Transfer Function with Consistent Internal Scaling.	90
4.8	Overall System Scaling.	93
5.1	Graphical Solution of Eq.(3.52) for Plant A.	109
5.2	Closed-Loop Unit Step Responses of Plant A for Various LQG Weighting Factors ρ.	110

5.3 Performance Estimate Eq.(5.10) for Plant A with b=4 and 3σ Loading. 112

5.4 Steady-State Mean-Square-Error for ρ=1 and Various C values. (a)b=4 Bits. (b)b=3 Bits. 113

5.5 Steady-State Mean-Square-Error for ρ=10 and Various C values. (a)b=4 Bits. (b)b=3 Bits. 114

5.6 Steady-State Mean-Square-Error for $\rho=10^2$ and Various C values. (a)b=4 Bits. (b)b=3 Bits. 115

5.7 Steady-State Mean-Square-Error for $\rho=10^3$ and Various C values. (a)b=4 Bits. (b)b=3 Bits. 116

5.8 Steady-State Mean-Square-Error for $\rho=10^4$ and Various C values. (a)b=4 Bits. (b)b=3 Bits. 117

5.9 Steady-State Control System Performance for Various M with b=3 and an ADPCMQ Coder. 123

5.10 Steady-State Control System Performance for Various M with b=4 and an ADPCMQ Coder. 124

5.11 Plant Output and Step-Size Responses to a Nonstationary Process Disturbance and a Zero Set-Point (Regulator) for Time k=850 to k=1450 with an ADPCMQ Coder. (a)M=4. (b)M=2. 128

5.12 Plant Output and Step-Size Responses to a Nonstationary Process Disturbance and a Zero Set-Point (Regulator) for Time k=3350 to k=3950 with an ADPCMQ Coder. (a)M=4. (b)M=2. 129

5.13 Plant Output and Step-Size Responses to a Nonstationary Process Disturbance and a Constant Non-Zero Set-Point for Time k=850 to k=1450 with an ADPCMQ Coder. (a)M=4. (b)M=2. 130

5.14 Plant Output and Step-Size Responses to a Nonstationary Process Disturbance and a Constant Non-Zero Set-Point for Time k=3350 to k=3950 with an ADPCMQ Coder. (a)M=4. (b)M=2. 131

5.15 Steady-State Control System Performance

Comparison for ADPCMF and ADPCMQ Coders with b=3 and Various Quantizer Loadings. 136

5.16 Steady-State Control System Performance Comparison for ADPCMF and ADPCMQ Coders with b=4 and Various Quantizer Loadings. 137

5.17 Plant Output and Step-Size Responses to a Nonstationary Process Disturbance and a Zero Set-Point (Regulator) with an ADPCMF Coder. (a)Time k=850 to k=1450. (b)Time k=3350 to k=3950. 140

5.18 Plant Output and Step-Size Responses to a Nonstationary Process Disturbance and a Constant Non-Zero Set-Point with an ADPCMF Coder. (a)Time k=850 to k=1450. (b)Time k=3350 to k=3950. 141

5.19 Steady-State Control System Performance Comparison for Fixed A/D, ADPCMQ, and ADPCMF for ρ=1. 144

5.20 Steady-State Control System Performance Comparison for Fixed A/D, ADPCMQ, and ADPCMF for ρ=10. 145

5.21 Steady-State Control System Performance Comparison for Fixed A/D, ADPCMQ, and ADPCMF for $\rho=10^2$. 146

5.22 Steady-State Control System Performance Comparison for Fixed A/D, ADPCMQ, and ADPCMF for $\rho=10^3$. 147

5.23 Steady-State Control System Performance Comparison for Fixed A/D, ADPCMQ, and ADPCMF for $\rho=10^4$. 148

5.24 Steady-State Control System Performance of Fixed A/D for b=4 and Various LQG Weighting Factors ρ. 151

5.25 Steady-State Control System Performance of ADPCMQ Coder for b=4 and Various LQG Weighting Factors ρ. 152

5.26 Steady-State Control System Performance of ADPCMF Coder for b=4 and Various LQG Weighting Factors ρ. 153

5.27 Plant Output and Step-Size Responses to Random Set-Points for $R_w=10^{-1}$. 156

5.28 Plant Output and Step-Size Responses to Random Set-Points for R_w=1. 157

5.29 Plant Output and Step-Size Responses to Random Set-Points for R_w=10. 158

5.30 Control System Performance Comparison for Fixed A/D, ADPCMQ, and ADPCMF Coders with ρ=1 and Random Set-Points. 159

5.31 Control System Performance Comparison for Fixed A/D, ADPCMQ, and ADPCMF Coders with ρ=10 and Random Set-Points. 160

5.32 Control System Performance Comparison for Fixed A/D, ADPCMQ, and ADPCMF Coders with $\rho=10^2$ and Random Set-Points. 161

5.33 Control System Performance Comparison for Fixed A/D, ADPCMQ, and ADPCMF Coders with $\rho=10^3$ and Random Set-Points. 162

5.34 Control System Performance Comparison for Fixed A/D, ADPCMQ, and ADPCMF Coders with $\rho=10^4$ and Random Set-Points. 163

5.35 Control System Performance of Fixed A/D for b=4 and Various LQG Weighting Factors ρ with Random Set-Points. 166

5.36 Control System Performance of ADPCMQ Coder for b=4 and Various LQG Weighting Factors ρ with Random Set-Points. 167

5.37 Control System Performance of ADPCMF Coder for b=4 and Various LQG Weighting Factors ρ with Random Set-Points. 168

5.38 Graphical Solution of Eq.(3.52) for Plant B. 170

5.39 Closed-Loop Unit Step Responses of Plant B for Various LQG Weighting Factors ρ. 170

5.40 Steady-State Mean-Square-Error of Plant B for Various c Values. 171

5.41 Comparison of Steady-State Mean-Square-Error of Plant B for Fixed A/D, ADPCMQ, and ADPCMF. 173

5.42 Plant Output and Step-Size Responses of Plant B to a Very Large Process Disturbance. (a)ADPCMQ, $R_w=1.20\times10^5$. (b)ADPCMF, $R_w=2.5\times10^5$. 175

5.43 Comparison of Mean-Square-Error of Plant B for Fixed A/D, ADPCMQ, and ADPCMF with Random Set-Points. 177

A.1 LQG Control System with Non-Zero Set-Point [After Moroney (1983)]. 194

Chapter 1

INTRODUCTION

1.1 Background

It is well known that the inherent errors due to finite wordlength effects can affect the performance of a digital feedback control system which is implemented with finite wordlength hardware components. In some cases, inadequate wordlength may even cause instability of the closed-loop system. The main sources of these finite wordlength effects are the round-off errors in the arithmetic processors used to realize the control system, quantization error of the controller parameters when they are represented by computer memory words with finite accuracy, the quantization error introduced by analog-to-digital (A/D) converters, and digital-to-analog (D/A) conversion errors.

Finite wordlength effects have aroused great interest and been studied extensively in the communication and signal processing areas (Oppenheim and Schafer, 1975; Rabiner and Gold, 1975). Many useful theory and ideas are available in these areas. But, they are not directly applicable to a digital feedback control system. The reason is that, in control systems, the presence of a feedback path will couple the plant output back to the input through the D/A and A/D converters. In communication and signal processing, however, such feedback coupling effect is not present and hence the analysis of the finite wordlength effects would be different

from that in control systems. Finite wordlength effects have also been studied in the context of a closed-loop feedback system (Bertram, 1958; Slaughter, 1964; Knowles and Edwards, 1965). More recent thorough investigations of finite wordlength effects in control systems are that of Rink and Chong (1975, 1979). The most recent results are that of Moroney (1983) and van Wingerden and De Konig (1984).

An important finite wordlength problem which has not received much attention in the control systems area is that of the conflicting requirements between the A/D conversion error, commonly called quantization noise, and the A/D converter dynamic range. Consider a typical input-output characteristic of an A/D converter, or equivalently, a fixed quantizer, as shown in Fig. 1.1. As is evident from Fig. 1.1, the input signal can be approximated by a set of finite discrete quantizer output levels. Thus only a finite dynamic range of input signals can be approximated by the quantizer without introducing much error. Given an acceptable quantization error level and hence a given quantization step-size, if a wide dynamic range is desired, it is necessary to increase the number of quantizer output levels, or the number of quantizer bits. On the other hand, given the number of quantizer bits, the dynamic range will depend on the allowable quantization error level. If only low quantization error is acceptable, a small dynamic range will be implied; however, if a wide dynamic range is needed, a large quantization error will result. In many

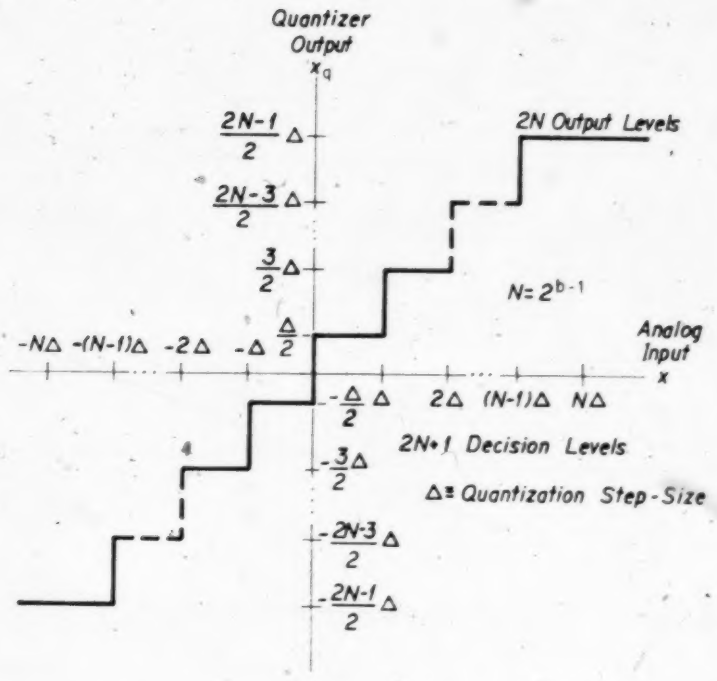


Fig. 1.1 Input-Output Characteristic of a Uniform b-bit Mid-Riser Quantizer.

applications, this conflict is simply solved with very conservative design using very long A/D wordlengths. But, this approach is not very cost-effective. In cases where the A/D wordlength is restricted, such as when fast A/D converters which usually have very short wordlengths are required, this approach of design will not be feasible.

Though this problem has not been studied extensively in the control systems area, it has received much attention in the communication area and many useful techniques for efficient coding have resulted. One particular scheme is known as Adaptive Differential Pulse Code Modulation (ADPCM) coding for purposes of bit-rate reduction in speech coding

and communication applications. This coding scheme is found to be very efficient and is capable of extending the useful dynamic range of the coder even when the A/D wordlength is very limited. Applications of such a coding scheme to speech signals have been studied extensively via computer simulations and hardware implementations. Results have shown that this class of coding scheme has superior performance to that of conventional coding schemes. Comprehensive reviews on ADPCM systems can be found in Jayant (1974), Rabiner and Schafer (1978), and Gibson (1980a).

In view of the success in applying ADPCM coding techniques in communication, it is believed that similar benefits can also be obtained when such coders are used as A/D converters in feedback control systems. Rink (1982) has recently attempted an application of a nonadaptive version of this coding technique, as an efficient quantization scheme, by applying a nonadaptive Differential Pulse Code Modulation (DPCM) coding scheme in the feedback path of a localized control system and has found that it can provide significant improvement in control system performance over that of the conventional fixed quantization schemes. Fischer and Tinnin (1982, 1984), in an attempt to solve the problem of joint optimization of control and communication, have also made use of an ADPCM coder in a centralized LQG regulator problem. They have also observed similar performance improvements.

1.2 Objective and Scope of Thesis

Based on the promising results of the earlier attempts made by Rink (1982) and Fischer and Tinnin (1982, 1984), the ADPCM coding technique will be applied to a control system. The main concern of this thesis is to study the benefits of an ADPCM coder in a localized control system and the research is oriented to practical hardware implementation considerations, rather than the purely theoretical approach of Fischer and Tinnin (1982, 1984). This is motivated by the fact that many important issues in hardware implementations of digital control systems are still unexplored and unanswered. It is expected that the results of this research would be very useful to serve as a modest starting point to a vast unexplored area, which is the search for better quantization methods for control systems. The ideas presented in this thesis may also be very useful in giving insights to digital control system designers who need to implement their designs with practical hardware components.

This study is essentially an extension of the ideas of Rink (1982) from a nonadaptive version to an adaptive version of the DPCM quantization scheme, with emphasis on practical implementation considerations. Since the problem is concerned with digital control, the study is limited to discrete-time systems. Thus the system is assumed to have been sampled properly at the appropriate locations.

With a time-varying and signal-dependent ADPCM coder in the control loop, the system becomes highly nonlinear.

Closed form analytical solutions from theoretical analysis may not be obtainable. Hence, this study will make use of a simplified, approximate approach of linear analysis and will rely on extensive computer simulations, performed on two particular example systems, for specific quantitative results. These simulation results and the results from the approximate analysis can then be used to evaluate the performance of the proposed system, when compared with a system using conventional fixed A/D conversion methods. In the simulations two different types, the feedforward and the feedback type, of ADPCM coders are applied and their contributions to the improvements in control system performance are also compared.

1.3 Outline of Thesis

Before the proposed control system is presented, a brief review of differential encoding systems in waveform communication is given in Chapter 2. Then, in Chapter 3, the configuration of the proposed system is discussed. Design methods for the equations used in the control system as well as for the quantizer used in the ADPCM coder are also given in this chapter. It is followed by Chapter 4 which contains a brief discussion on the considerations of hardware implementations of the control system and suggestions for realization with practical hardware components are also included. The main results of this research, based on the extensive computer simulation results, are presented

graphically and discussed in Chapter 5. Finally, summary and conclusions of this thesis, together with a few suggestions for further study, are given in Chapter 6.

Chapter 2

ADAPTIVE DIFFERENTIAL QUANTIZATION IN DIGITAL WAVEFORM COMMUNICATION

2.1 Introduction

The function of a digital waveform communication system is the transmission of analog waveforms, as accurately as possible, in a digital format via a digital communication channel or communication link. At the receiving end of the channel, the original analog waveforms are reconstructed from the digital data received. Before the analog waveforms can be transmitted, they must be converted from an analog format into a digital format. This requires analog-to-digital (A/D) converters to perform the conversion, which involves a sampling and a quantization process. The analog waveforms must first be converted from a continuous-time format into a discrete-time format by sampling the waveforms at discrete time intervals. These signal samples, whose amplitudes may assume any of a continuum of values ranging from $-\infty$ to $+\infty$ (Gersho, 1978), are quantized into samples having only a finite set of discrete amplitude levels and then coded into digital data, of which binary numbers are the most widely used in practice. During the quantization, some information in the original signal samples is discarded. Therefore the reconstructed waveforms at the receiver will not be identical to the original waveforms but only an

approximation and a distorted version of the original. The distortion results from the errors introduced by the quantization process. Apart from possible communication channel errors, the quantization error is a fundamental limitation of the digital waveform communication system. Thus a well designed system should be one that contains as little quantization error as possible.

In order to minimize the quantization error, one may use a large number of binary digits (bits) so that a very large set of discrete amplitude levels is available for representing and coding the waveform samples. This is, however, not an efficient way of designing the communication system since a large number of bits implies a high data transmission bit-rate. This will require a large transmission channel capacity or transmission bandwidth and will lead to an increase in the cost of transmission as well. Thus a conflicting requirement between the quantization accuracy and the data transmission bit-rate arises. In practice, it is often desirable to minimize the channel capacity requirements by reducing the transmission bit-rate so that the cost of transmission can also be minimized. This reduction of bit-rate is usually achieved at the expense of an acceptable amount of quantization error distortion in the output waveforms at the receiving end of the communication channel (Davisson, 1972).

In order to achieve bit-rate reduction, it is necessary to reduce the number of data bits used for coding the signal

samples. This requirement may, however, create another problem. Given a certain acceptable level of quantization error, reducing the number of the data bits used to represent the signal samples implies a decrease in the dynamic range of signal samples that can be accommodated. Alternatively, to maintain the desirable wide dynamic range, a large quantization step-size is required. This may result in a significant loss of the original information due to the coarse quantization. Thus a conflicting requirement between the dynamic range of the signal samples that can be accommodated by the quantizer and the quantization accuracy arises. To solve this problem a more efficient way of quantization must be found.

It is well known that when speech or image signals are sampled at the Nyquist rate, which is twice the highest frequency component in the waveforms, there exists a significant amount of redundancy in the signal samples in the form of intersample correlation (Jayant, 1974). It is found that direct coding, which is a widely used method of quantization in practice, of these signal samples by means of memoryless quantizers such as uniform quantizers, logarithmic companded quantizers, or optimum quantizers, is inadequate and inefficient (Gersho, 1978) since a large number of bits is wasted in coding the redundant information. If the redundancy can be removed from the signal samples before quantization, it is possible to reduce the number of bits required for coding the information. This

is exactly the procedure used in a predictive coding system (Elias, 1955; O'Neal, 1966a; Atal and Schroeder, 1970; Gibson, 1980a) which makes use of the intersample correlation, based on past information of the signal samples, to reduce the signal redundancy. It has been successfully applied to speech and image signals and is found to be very efficient and effective in realizing the objective of bit-rate reduction and data compression (O'Neal, 1966a; Atal and Schroeder, 1970; Jayant, 1974, 1976; Noll, 1975a; Gibson, 1980a). Since past information of the signal samples is required in the removal of the redundancy, predictive coding systems are generally classified as quantizers having memory as compared with the memoryless quantizers mentioned before. The most common form in which predictive coding is applied is the differential encoding system which includes schemes such as Differential Pulse Code Modulation (DPCM) and Delta Modulation (DM) as well as their adaptive versions (Jayant, 1974; Gibson, 1980a).

In the following sections, the basic principle of differential encoding systems and their adaptive versions will be described briefly. More detailed discussion on the subject of predictive coding is that of Elias (1955) and comprehensive reviews on the subject of differential encoding can be found in Jayant (1974), Rabiner and Schafer (1978), and Gibson (1980a).

2.2 Differential Encoding System

2.2.1 Basic System Description

The basic general configuration of a differential encoding communication system is shown in Fig. 2.1 in a block diagram form. At the transmitter, the predictor P, based on information derived from past information of the input waveform samples $\{y(k-1), y(k-2), \ldots\}$, generates a predicted value of the current input sample $y(k)$. This predicted value, $\hat{y}(k|k-1)$, is first subtracted from the input to remove the signal redundancy so that only the information which is essential for reproducing the signal is retained. The resulting difference signal, or the prediction error,

$$\delta(k) = y(k) - \hat{y}(k|k-1), \qquad (2.1)$$

is then quantized, encoded, and transmitted as channel code-words to the receiver via the communication channel. The advantage of removing the redundancy is that the variance of the signal to be quantized is reduced and hence the dynamic range is also reduced (Noll, 1975a). The quantized prediction error, $\delta_q(k)$, is added back to the predicted value to reconstruct the original signal sample. This reconstructed signal $\hat{y}(k|k)$, a quantized version of the original, then becomes part of the past information to be used by the predictor in the feedback loop to form the next predicted value $\hat{y}(k+1|k)$. It should be noted, however, that

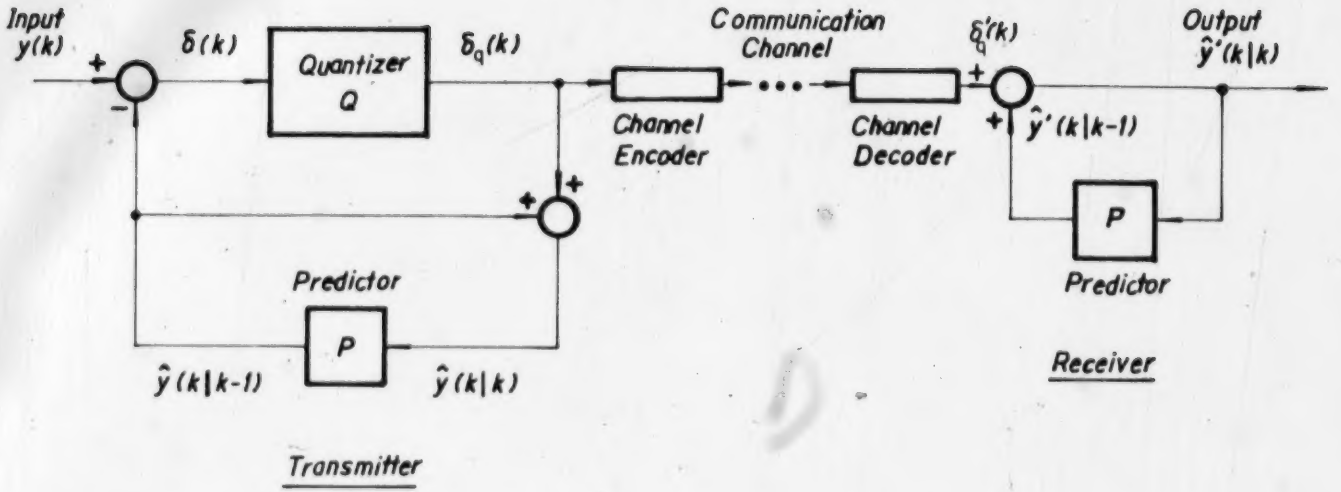


Fig. 2.1 Differential Encoding Communication System.

$\hat{y}(k|k)$ is only an approximation to the original sample because of the inevitable error introduced in the process of quantizing $\delta(k)$:

$$\delta_q(k) = \delta(k) + \epsilon(k), \tag{2.2}$$

where $\epsilon(k)$ represents the quantization error at time k. Generally, $\epsilon(k)$ is assumed to be an additive zero-mean, white, and uniformly distributed random noise sequence. For uniform quantization,

$$|\epsilon(k)| \leq \frac{\Delta}{2} = \frac{1}{2} \cdot \frac{\delta_{max} - \delta_{min}}{2^b}, \tag{2.3}$$

where Δ is the quantizer step-size and b is the number of quantizer bits. The difference between $\hat{y}(k|k)$ and $y(k)$ is simply the quantization error (Atal and Schroeder, 1970):

$$\begin{aligned} \hat{y}(k|k) &= \hat{y}(k|k-1) + \delta_q(k) \\ &= \hat{y}(k|k-1) + \delta(k) + \epsilon(k) \\ &= y(k) + \epsilon(k). \end{aligned} \tag{2.4}$$

At the receiver, the removed redundancy must be reinserted (Gibson, 1980a) to reconstruct the original input sample. This is done by repeating the identical reconstruction process as in the transmitter. The transmitted signal is decoded into $\delta_q'(k)$ and then added to the predicted value $\hat{y}'(k|k-1)$, generated by the predictor P at time $k-1$, to form the reconstructed signal $\hat{y}'(k|k)$. This

then becomes part of the past information to be used by P in the receiver predictor feedback loop to generate the next predicted value $\hat{y}'(k+1|k)$. It is evident from Fig. 2.1 that, if no channel transmission errors are present, $\delta_q''(k)$, $\hat{y}'(k|k)$, and $\hat{y}'(k|k-1)$ at the receiver are identical to $\delta_q(k)$, $\hat{y}(k|k)$, and $\hat{y}(k|k-1)$, respectively, at the transmitter. Again $\hat{y}'(k|k)$ differs from $y(k)$ only by the quantization error $\epsilon(k)$ (Atal and Schroeder, 1970).

2.2.2 Signal-to-Quantization Noise Ratio Improvement

In the absence of transmission channel errors, the signal-to-quantization noise ratio (SNR) of the reconstructed signal at the receiver output is, in standard statistical notation,

$$SNR = \frac{\langle y^2(k)\rangle}{\langle \epsilon^2(k)\rangle}, \tag{2.5}$$

where $\langle\bullet\rangle$ indicates a time averaging operation. Alternatively, this can be written as (Jayant, 1974; Rabiner and Schafer, 1978)

$$SNR = \frac{\langle y^2(k)\rangle}{\langle \delta^2(k)\rangle}\frac{\langle \delta^2(k)\rangle}{\langle \epsilon^2(k)\rangle} = G_p \cdot SNRQ, \tag{2.6}$$

where the SNR of the quantizer is given by

$$SNRQ = \frac{\langle \delta^2(k)\rangle}{\langle \epsilon^2(k)\rangle}, \tag{2.7}$$

and the gain due to the differential configuration is defined as

$$G_d = \frac{\langle y^2(k) \rangle}{\langle \delta^2(k) \rangle}. \qquad (2.8)$$

For uniform quantization and a b-bit quantizer

$$\langle \epsilon^2(k) \rangle = \frac{\Delta^2}{12} = \frac{1}{12} \cdot \frac{(\delta_{max} - \delta_{min})^2}{2^{2b}}. \qquad (2.9)$$

As can be seen from Eq. (2.6), two factors contribute to the SNR of the reconstructed signal. For a given quantizer with given parameters, *SNRQ* is fixed. For speech waveform signals, G_d is often found to be much greater than unity and can usually offer a gain of about 6 dB up to about 16 dB, depending on the form of the predictor used (Noll, 1975a). Thus the SNR is greatly increased and this results in a system performance better than that obtained by applying the same quantizer directly to the input samples. The SNR improvement is a result of the differential configuration which is essentially a quantization noise reduction technique using a feedback loop around the quantizer (Spang and Schultheiss, 1962). On the other hand, the differential configuration permits a reduction of the quantizer wordlength. This is a result of the dynamic range reduction by the removal of signal redundancy in the signal samples before quantization. When the dynamic range of the signal to be quantized is reduced, one can decrease the number of

quantization levels by reducing the number of quantizer bits. This lowers the *SNRQ*, but the decrease is compensated by the gain in G_d due to the differential configuration and hence a specified SNR performance can still be maintained. The reduction of quantizer wordlength thus makes bit rate reduction, and hence bandwidth reduction possible. Consequently, the existing transmission and storage facilities can be used in a more efficient way.

2.2.3 Predictor

The function of the predictor *P* in Fig. 2.1 is, basically, to generate an estimate of the future input sample values using available information about the past input samples. The predictor estimate is then subtracted from the input sample to reduce the signal redundancy in the signal to be quantized. The most common approach of prediction is by means of the linear prediction method (Elias, 1955; O'Neal, 1966a; Atal and Schroeder, 1970; Jayant, 1974; Makhoul, 1975).

The input speech signal is modelled as a linear combination of *M* past signal samples:

$$y(k) = \sum_{i=1}^{M} a_i y(k-i). \qquad (2.10)$$

The prediction is then generated by a linear combination of *M* past reconstructed samples:

$$\hat{y}(k|k-1) = \sum_{i=1}^{M} a_i \hat{y}(k-i|k-i). \qquad (2.11)$$

This predictor structure is essentially a transversal filter having M tapped delay lines weighted by the coefficients a_i (Rabiner and Gold, 1975). From Eq. (2.6), for a given SNRQ and a fixed input power level $\langle y^2(k) \rangle$, the output SNR can be maximized by maximizing the gain G_p. This requires the variance of the prediction error $\langle \delta^2(k) \rangle$, given by

$$\begin{aligned} \langle \delta^2(k) \rangle &= \langle [y(k) - \sum_{i=1}^{M} a_i \hat{y}(k-i|k-i)]^2 \rangle \\ &= \langle [y(k) - \sum_{i=1}^{M} a_i y(k-i) - \sum_{i=1}^{M} a_i \epsilon(k)]^2 \rangle, \end{aligned} \qquad (2.12)$$

to be minimized by the appropriate selection of the coefficients a_i in Eq. (2.12). Detailed discussion on how these coefficients can be obtained is given by Atal and Schroeder (1970), Jayant (1974), and Rabiner and Schafer (1978). One may also notice that the quantization error will affect the performance of the predictor as well because the reconstructed signal contains the quantization error. Results have shown that a gain of about 6 dB in G_p is attainable even with a first order (M=1) predictor. Further increasing the predictor order M does not increase this gain significantly for speech signals (Jayant, 1974; Noll, 1975a; Rabiner and Schafer, 1978). Hence, the first order predictor has widely been used in practice in differential encoders because of its simplicity of implementation and its quite

acceptable gain in G_d (O'Neal, 1966b; Cummiskey *et al.*, 1973; Goldstein and Liu, 1977a, 1977b; Dubnowski, 1978; Adelmann *et al.*, 1979; Un and Cynn, 1980; Boddie *et al.*, 1981).

It is possible to further increase the output SNR of the differential encoding system by using a Kalman filtering approach to design the predictor (Gunn and Sage, 1973; Melsa and Kolstad, 1977; Gibson, 1980b). A Kalman filter is desirable because it is capable of yielding optimum least squares estimates of the signal samples. Furthermore, it is also capable of suppressing the quantization noise in the predictor feedback loop whereas the transversal filter type predictor does not have such feature. Results have shown that the Kalman filtering technique is very effective in reducing quantization noise and the SNR performance is better than those systems having only a transversal filter predictor in the feedback loop (Melsa and Kolstad, 1977; Gibson, 1980b).

It is well known that the statistics of speech signals are nonstationary and vary widely for different speakers. Usually, only a limited amount of the knowledge of these statistics is available (Noll, 1975a). Predictors having fixed coefficients or parameters will not suffice in dealing with the nonstationarity and the inexact knowledge of the statistics of the signals. The obvious solution is to make the predictors adaptive to match the time-varying speech signal statistics. The parameters or coefficients of the

predictors can be updated peribdically by means of sophisticated parameter estimation schemes, such as steepest descent gradient search technique (Jayant, 1974) and Kalman filtering technique (Gibson *et al.*, 1974; Gibson, 1978, 1980a, 1980b), using past reconstructed signal sample. Typically, an increase of 2-4 dB in G_e over that of the fixed predictor is attainable (Gibson, 1980a). However, this minimal improvement usually does not justify the complex hardware implementation of the adaptive predictor algorithms. In practice, fixed predictors are considered to be adequate and satisfactory.

2.2.4 Quantizer

Although a memoryless quantizer is inefficient in direct coding of speech signal samples, it is still an indispensible element in the differential encoding system. Once the redundancy is removed or reduced, a memoryless quantizer can then be applied efficiently. Selection of a suitable quantizer depends on the encoding scheme to be used. If Differential Pulse Code Modulation (DPCM) coding is desired, one would choose a multi-bit quantizer. For a Delta Modulation (DM) coding scheme, one would then select a two-level quantizer. For a two-level quantizer, one only need to select a step-size that will yield the specified SNRQ. For a multi-bit quantizer, there are several alternatives one can consider.

Uniform quantizers, in which the quantization level spacings are distributed uniformly, are simple to implement. The disadvantage is, however, that the SNRQ is very sensitive to the input power level and a specified SNRQ cannot be maintained over a significant dynamic range (Gersho, 1978). Consequently, this limits the range of power level variations that can be accommodated. This problem can be remedied quite readily by means of nonuniform quantization in which the quantization level spacings are distributed nonuniformly (Gersho, 1978).

The most common nonuniform quantization technique is the logarithmic companding method, or commonly called μ-law companding quantization. The input samples are first "compressed" by a nonlinear function, commonly known as the μ-law, such that samples having low amplitude will occupy more quantization levels while those having high amplitude will occupy fewer. The compressed samples are then uniformly quantized. By "expanding" the quantized samples with an inverse function of the nonlinear function used in the compression, the original information can be recovered with a high degree of fidelity. The advantage offered by this scheme is that a specified SNRQ can be maintained over a dynamic range significantly wider than that of a uniform quantizer having the same number of quantization levels at the same specified SNRQ (Gersho, 1978).

Given the probability distribution function (PDF) of the input samples, it is possible to design a nonuniform

quantizer that is able to match this PDF to yield an optimum SNRQ. Max (1960) assumes a Gaussian PDF and obtains the parameters for the optimum quantizer. Paez and Glisson (1972) assume a Laplacian PDF to approximate a Gamma distribution, which resembles closely to the speech signal PDF, and optimum quantizer parameters for such a PDF have also been obtained. Results have shown that, for a given fixed input power level, when the quantizers are matched to the input PDF optimum quantizers can yield a higher SNRQ than that of logarithmic companded quantizers (Rabiner and Schafer, 1978). However, the same dynamic range problem as in uniform quantization also applies to optimum quantizers. This is because they are only optimized for a particular input signal PDF and power level. When these vary, the quantizers are no longer matched to the statistics of the signal. Hence, because of its wide dynamic range, a logarithmic companded quantizer is more favourable than an optimum quantizer in spite of an inferior SNRQ (Rabiner and Schafer, 1978).

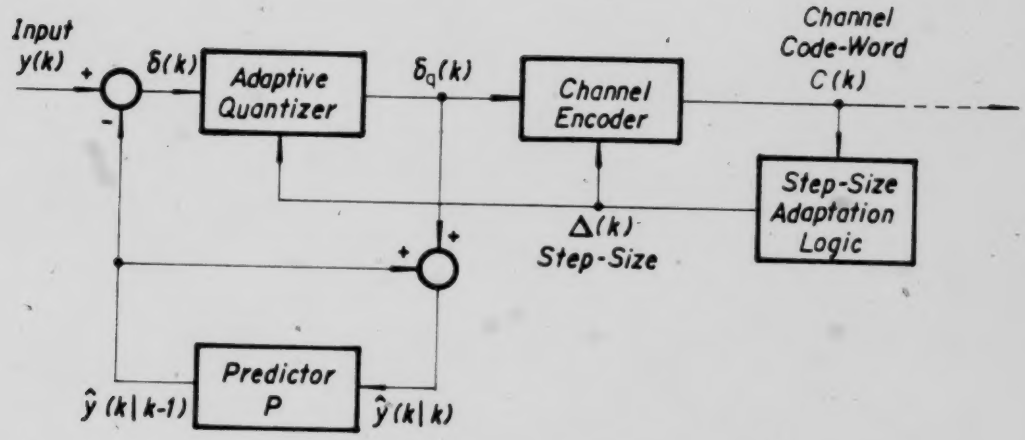
There exists, however, a wide variation of power levels in speech signals for different speakers. The fact that the power level is not always known has motivated the design of adaptive quantizers (Jayant, 1973; Mitra, 1974; Goodman and Gersho, 1974; Noll, 1974; Goodman and Wilkinson, 1975). The basic idea is to increase the quantization step-size when it is evident that the amplitude of the input samples are large and vice versa. Thus, by adapting the step-size, a much

wider dynamic range can be accommodated than that of the conventional quantization techniques. When used in conjunction with differential encoding systems, a much more significant improvement in system performance can be expected.

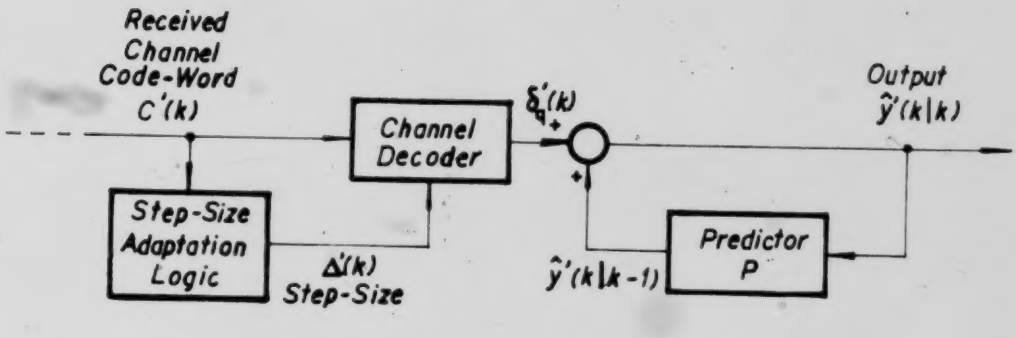
2.3 Adaptive Differential Encoding System

When the quantizer in Fig. 2.1 is replaced with an adaptive quantizer, an adaptive differential encoding system results. It includes the adaptive versions of DPCM and DM, called Adaptive DPCM (ADPCM) and Adaptive DM (ADM), respectively (Song *et al.*, 1971; Jayant, 1970, 1974; Cummiskey *et al.*, 1973). Basically, it combines the advantages of differential encoding systems (redundancy removal) and adaptive quantization (accommodation of wide dynamic range) (Cummiskey *et al.*, 1973). When the predictor in ADPCM or ADM systems is also made adaptive, as discussed earlier, to account for the nonstationarity and inexact knowledge of the speech signal statistics, ADPCM and ADM systems can yield a very efficient coding method with a significant improvement in system performance. Comprehensive reviews on the subject of differential encoding and adaptive prediction in speech coding can be found in Jayant (1974), Rabiner and Schafer (1978), and Gibson (1980a).

Two typical ADPCM system configurations are shown in Figs. 2.2 and 2.3. Basically, they are identical to Fig. 2.1 except that extra blocks of step-size adaptation logic are



(a) Transmitter



(b) Receiver

Fig. 2.2 ADPCM System with Feedback Adaptive Quantizer [After Rabiner and Schafer (1978)].

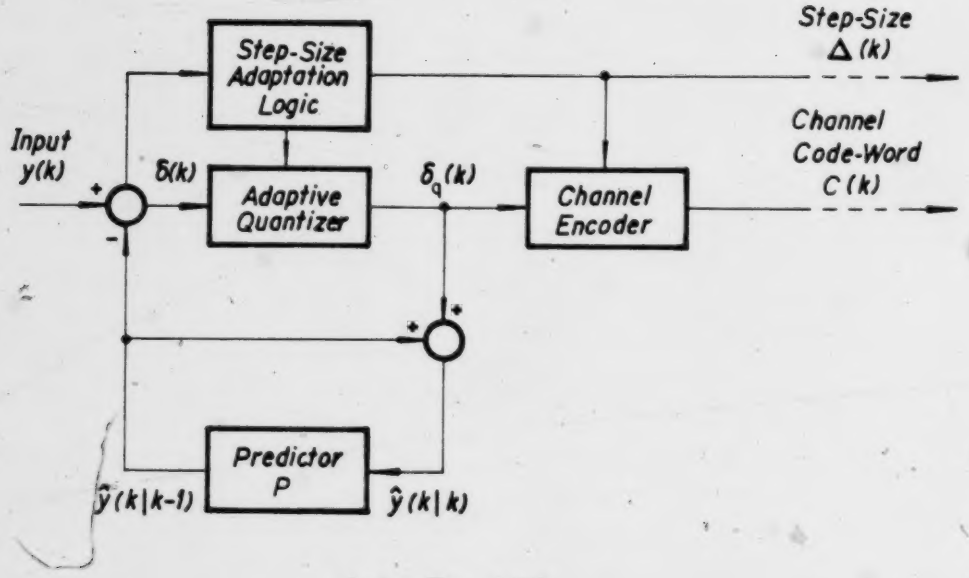


Fig. 2.3 ADPCM System with Feedforward Adaptive Quantizer [After Rabiner and Schafer (1978)].

included. The function of the step-size adaptation logic is to update the quantizer step-size $\Delta(k)$ at each sample time k so that a suitable $\Delta(k)$ would be used to quantize the current input signal which is the prediction error $\delta(k)$. This step-size information must also be transmitted to the receiver so that a consistent step-size is used by the transmitter and the receiver when encoding and decoding the transmitted code-word. The method of transmitting this information depends on whether a feedback or a feedforward type of adaptive quantizer is used in the ADPCM transmitter (Noll, 1975a; Rabiner and Schafer, 1978).

In the ADPCM system having a feedback adaptive quantizer (ADPCMQ) as shown in Fig. 2.2, the step-size adaptation logic is controlled by the channel code-word sequence $C(k)$ which is related to the quantizer output $\delta_q(k)$. At each sample time k, the adaptation logic derives information about the step-size from the code-word sequence $C(k)$ and updates the step-size $\Delta(k)$ accordingly. This step-size is used to quantize the prediction error $\delta(k)$ into $\delta_q(k)$ which is then encoded into a channel code-word $C(k)$ and transmitted to the receiver. At the receiver, an identical adaptation logic uses the received channel code-word sequence $C'(k)$ to reconstruct the step-size information $\Delta'(k)$. This derived information is then used in the decoding of $C'(k)$ to retrieve $\delta'_q(k)$ from which $\hat{y}'(k|k)$ can be reconstructed. Since the step-size information is already contained in the channel code-word sequence in the

feedback quantizer system, additional step-size information $\Delta(k)$ need not be transmitted to the receiver.

In the ADPCM system having a feedforward adaptive quantizer (ADPCMF) as shown in Fig. 2.3, the step-size adaptation logic is controlled directly by the quantizer input sequence $\delta(k)$ rather than by the information derived from the channel code-word sequence $C(k)$ as in the ADPCMQ system. At each sample time k, the adaptation logic updates the step-size $\Delta(k)$ from information in the input $\delta(k)$ sequence. The quantizer then uses this step-size to quantize $\delta(k)$ to obtain $\delta_q(k)$ which is then coded and transmitted to the receiver. Since the $\delta(k)$ sequence is not available at the receiver, it is not possible for the receiver to decide what step-size $\Delta(k)$ is used by the transmitter at time k. In order to decode the received channel code-word $C'(k)$ correctly, it is necessary to transmit the step-size $\Delta(k)$ to the receiver in addition to the transmitted code-word. With the step-size transmitted explicitly the step-size adaptation logic is not needed at the receiver.

The advantage of both the feedback and the feedforward adaptive quantizer is their ability to adapt the step-sizes to a wide dynamic range of input signal power level variations. For speech signals, it is possible to attain a gain of about 4-6 dB in SNRQ over a standard μ-law non-adaptive quantizer having the same number of bits. The feedforward adaptive quantizer can yield a gain of about 5 dB in SNRQ over the non-adaptive μ-law quantizer with the

same number of bits (Noll, 1975a; Rabiner and Schafer, 1978). Combining a possible 6 dB gain due to the differential configuration having a first order predictor (Noll, 1975a; Rabiner and Schafer, 1978), a net 10-12 dB gain in the SNR is possible.

One should note, however, that the average quality of the reconstructed signal $\hat{y}'(k|k)$ greatly depends on the rate of transmission errors present in the code-words $C'(k)$. This is especially important in the ADPCMQ system because the receiver requires an accurate code-word sequence to faithfully reproduce the information $\Delta(k)$ and $\delta_q(k)$. The ADPCMF system has the advantage that it is not as susceptible to the channel error as is the ADPCMQ system because of the explicitly transmitted step-size information. It may, however, require extra bits and channel capacity for transmitting the step-size information, and the original objective of bit-rate reduction may be affected. Furthermore, a more complex circuitry may be required for a feedforward quantizer. With proper error protective codings, both the feedforward and the feedback quantizers in an ADPCM system can provide a desirable and efficient coding scheme for speech waveform coding and communication.

A DM coder can also be made adaptive by adapting the step-size of the quantizer. Either feedforward or feedback quantizers as in ADPCM coders can be used. Since only a one-bit code-word is involved in an ADM coder, it may be necessary to oversample the signal at a rate much higher

than the Nyquist rate to compensate for the coarse quantization. But because of its simplicity of implementation (Jayant, 1974; Rabiner and Schafer, 1978), ADM coders are still worth considering and are competitive to ADPCM coders.

Adaptive differential encoders for speech coding have been studied extensively and have been compared to other conventional coding schemes for digital waveform communication. Cummiskey *et al.* (1973) have compared, using computer simulations, the SNR of an ADPCM coder with that of a logarithmic companded coder (log-PCM) at different bit-rates. It is found that ADPCM can yield better performance than log-PCM at the same bit-rate. The ADPCM coder is also implemented with hardware components and evaluated, using speech signals, by subjective listening tests. Results have also shown that ADPCM gives a better subjective performance and is preferrable in many cases (Cummiskey *et al.*, 1973). Jayant (1974) compares an ADPCM speech coder with an ADM coder as well as a log-PCM in terms of SNR at different bit-rates. The comparison also reveals the superiority of ADPCM system over the others. ADM is also found to be superior to log-PCM at lower bit-rates, but relative performance deteriorates gradually when the bit-rate is increasing. At low bit-rates, the performance of ADM is only slightly inferior to that of ADPCM. It is suggested that ADM be considered at low bit rates because of the simplicity of its implementation (Jayant, 1974). Noll

(1975a) compares the performance of various quantization schemes for speech encoding and concludes that ADPCM with adaptive predictors are the best in terms of SNR performance. In a hardware implementation of a digital speech encoder, Johnston and Goodman (1978) compare the dynamic range of adaptive PCM (APCM), PCM, DPCM, and ADPCM. It is found that when the quantizer is made adaptive, a wider dynamic range can be accommodated than the range of the fixed quantizer. The differential configuration further enhances the SNR and gives an improvement of about 6 dB. Un and Cynn (1980) have also found that ADPCM and ADM can provide a fairly wide dynamic range. It has also been found, however, that the performance of ADPCM deteriorates rapidly in the presence of transmission channel errors (Noll, 1975b; Un and Cynn, 1980). This is because the removal of redundancy also removes the protection against the corruptive effects of channel errors in the transmitted signal (O'Neal, 1966a). Fortunately, this problem can be alleviated quite readily by using a robust adaptive quantizer which is capable of dissipating the effects of the transmission channel errors (Goodman and Wilkinson, 1975; Einarsson, 1981).

Generally speaking, it is found that ADPCM is more desirable than other coding schemes for speech waveforms because of its capability of increasing the dynamic range and reducing quantizer word-length while maintaining a specified SNR. Equivalently, given the number of quantizer

bits, ADPCM can provide a significant improvement of SNR over conventional direct PCM coding of speech signals while the useful dynamic range can be extended significantly.

Chapter 3

ADAPTIVE DIFFERENTIAL QUANTIZATION IN FEEDBACK CONTROL

3.1 Introduction

In view of the promising results offered by the ADPCM coding technique in digital waveform communication systems, one may apply an ADPCM coder as an A/D converter to a digital feedback control system and expect the same benefits from the coding method. Unlike the digital communication problem, where bit-rate and transmission bandwidth reductions are the main objectives, the consideration of applying an ADPCM coder to a control system is motivated by three other related factors. First, it is motivated by the distortion reduction and data compression feature of the differential encoding configuration. When this coder is applied to the feedback signal, the distortion would be reduced and hence better feedback information, for a given data rate, would be available to improve the control system performance. This may be very important, especially in cases where the A/D converter output wordlengths are very limited. For example, in applications where very fast quantization is required, one may need to employ very fast quantizers which are usually of very limited wordlengths. The benefits from the differential encoding configuration will generally improve the overall control performance in control systems where a separation theorem holds (Rink, 1982).

One can note quite readily that an ADPCM transmitter consists, basically, of a feedback loop around the quantizer and a predictor which generates an estimate of the input samples, to be subtracted from the input to form a residual signal which is to be quantized. If an ADPCM transmitter structure is implemented in the feedback path of a control system, the configuration will imply a separation theorem for control problems which is the second motivating factor. Curry (1970) proposed a separation theorem for an optimal stochastic control problem where an arbitrary nonlinear measurement device, such as an A/D converter, is operating in the feedback path of a linear controlled plant having a quadratic cost. It was demonstrated (Curry, 1970) that when the conditional mean is subtracted from the plant output via the measurement feedback loop, then the control signal would have no effects on the resulting residual which is to be quantized and thus cannot influence the quantizer distortion or the covariance matrix of state estimation errors. This leads to the separation property (Curry, 1970) in which the control action and state estimation can be computed separately. The main advantage of the separation property by using a feedback loop around the nonlinear measurement device is the reduction of the computational load and storage requirement required by a dynamic programming solution of the optimal stochastic control problem (Curry, 1970). Although nonlinear filters for generating the conditional mean may be needed, the separation property is

far more desirable than the dynamic programming approach since nonlinear filters may often be approximated or substituted with suboptimum filters. This separation property has also been applied to a joint optimization problem of control and communication (Fischer and Tinnin, 1982,1984) and to an efficient predictive quantization problem (Rink, 1982).

The third motivation is attributed to the benefits of adaptive quantization. By adapting the quantizer step-sizes, the useful dynamic range of the quantizer can be extended substantially and a consistent SNR can then be maintained quite readily over that range. Step-size adaptation is also useful when the quantizer input variance is unknown. By constantly adapting the step-sizes, it is possible to match the quantizer range to the unknown input variance so that the best possible quantizer performance can be obtained. Moreover, this technique of extending the quantizer dynamic range when the A/D converter output wordlength is limited, together with the efficient coding method of differential quantization, has been proven very effective and useful in the speech signal processing area (Jayant, 1974). On the whole, the application of an ADPCM transmitter in a digital feedback control system is motivated, basically, by the known benefits offered by the ADPCM coder in other applications. Similar benefits are to be expected in the performance of the control systems, especially when they are implemented with digital hardware which are limited by

finite computer wordlengths.

3.2 System Configuration and Description

The system under consideration, in which an ADPCM coder is used as an analog-to-digital (A/D) converter in a control loop, is shown in Fig. 3.1 in a block diagram form. It has a configuration similar to the system considered by Rink (1982) where a DPCM system is employed as the A/D converter in the feedback path of a localized digital feedback control system. In an ADPCM system, the fixed quantizer used in the DPCM system will be replaced with an adaptive one. Either a feedforward or a feedback adaptive quantizer can be used. For a localized control system application, both the channel encoder and decoder as well as the receiver side of the differential encoding system shown in Fig. 2.1 are normally not required. Only the transmitter side need be retained. In addition to the ADPCM transmitter in the feedback path, one must also include the blocks representing the controlled plant and the controller, denoted, respectively, by the pulse transfer functions $G(z)$ and $D(z)$, in the forward path to complete the control loop. This results in the basic configuration as shown in Fig. 3.1.

The controlled plant $G(z)$ (with zero-order-hold) is assumed to be a linear, time-invariant, single-input-single-output (SISO) dynamic process with $y(k)$ as its output. It is further assumed to be subject to an additive, stationary, zero-mean, white Gaussian plant disturbance

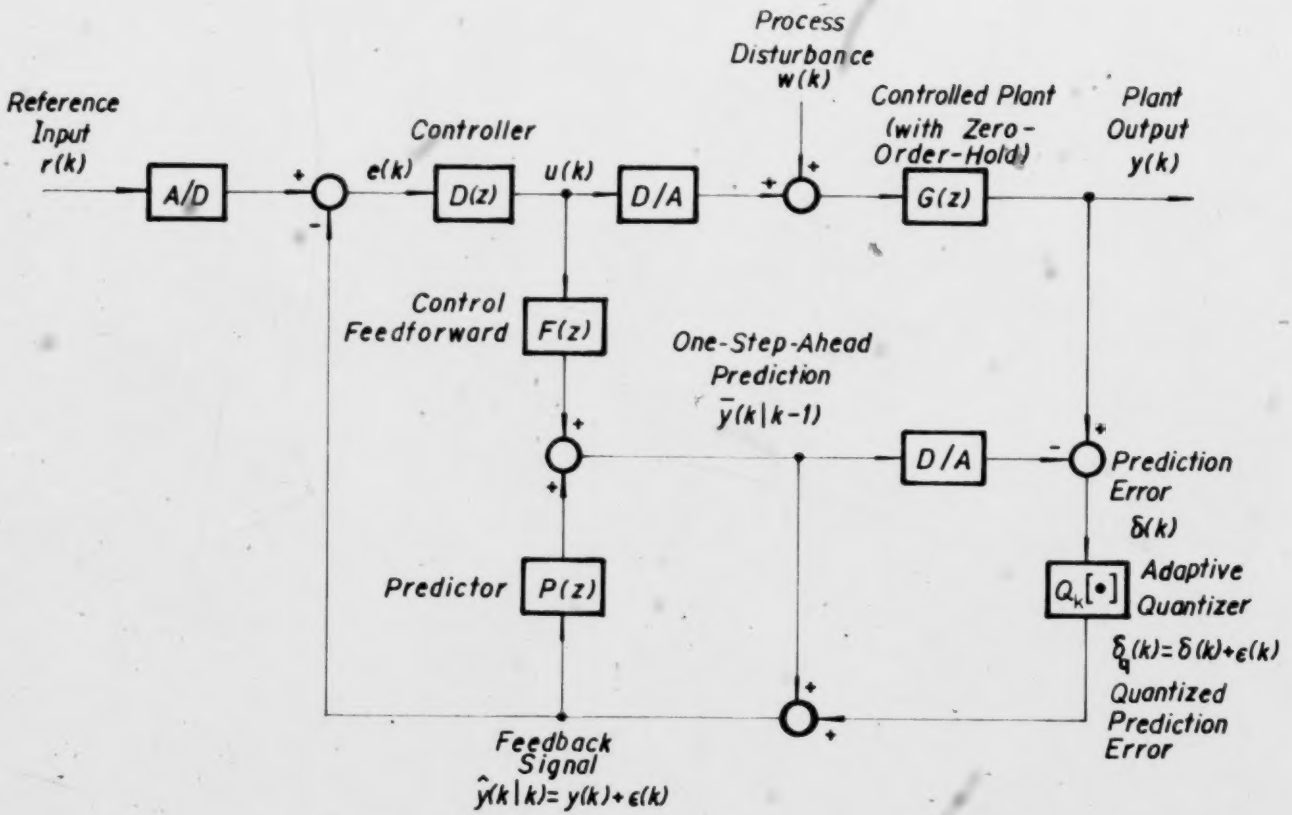


Fig. 3.1 Adaptive Differential Quantization Scheme in a Control Loop [After Rink (1982)].

$w(k)$. At each sampling time k, the controller $D(z)$, using the tracking error sequence $e(k)$ as its input, computes and generates the control signal $u(k)$ to be applied to the plant input. The controller $D(z)$ is to be designed to meet certain performance requirements for the plant output $y(k)$ dynamic response to the reference input $r(k)$.

One may notice that, in this unity-feedback control system, the plant output is not directly converted into digital form and is not fed back directly to form the tracking error $e(k)$. Instead, a one-step-ahead plant output prediction $\bar{y}(k|k-1)$, generated by the ADPCM predictor from past information about $y(k)$, is first subtracted from it to reduce its dynamic range and redundancy. The resulting prediction error $\delta(k)$ is then quantized by the adaptive quantizer $Q_*[\bullet]$ of the ADPCM transmitter. The quantized prediction error $\delta_q(k)$ is then added back to $\bar{y}(k|k-1)$ to reconstruct the original plant output sample $y(k)$. The reconstructed sample $\hat{y}(k|k)$ then becomes the actual feedback information about the plant output and is compared with the reference input sample $r(k)$ to generate the tracking error $e(k)$. It can be seen from Fig. 3.1 that $\hat{y}(k|k)$ is simply the decoded output of an ADPCM transmitter and, similarly, is comprised of the actual plant output $y(k)$ and the quantization error $\epsilon(k)$ introduced by the adaptive quantizer (Eq. (2.4)). Since the dynamic range of the quantizer input $\delta(k)$ is reduced, the quantization error $\epsilon(k)$ will be much smaller than that if the same quantizer is applied directly

to the plant output $y(k)$ which has a much wider dynamic range. With a smaller quantization error, the feedback information would be more accurate and hence better control performance can be expected.

In the case of a tracking control problem in which the plant is controlled in such a way that its output will follow a deterministic reference signal $r(k)$, information about the reference signal should also be used in forming the one-step-ahead plant output prediction $\bar{y}(k+1|k)$ for the next sampling time $k+1$. One way of including this information in the prediction is by introducing the control signal $u(k)$, which is related to the reference signal through the controller $D(z)$ and the tracking error $e(k)$, to the predictor through a feedforward transfer function $F(z)$ which is to be determined. Thus, it can be seen that the predictor $P(z)$, which operates on the reconstructed sequence $\hat{y}(k|k)$, and the feedforward transfer function $F(z)$ constitute the actual ADPCM predictor.

In Fig. 3.1, it has been assumed that the ADPCM A/D converter is introduced in the feedback path of the control loop to digitize the feedback information. This is quite an appropriate assumption since, in many cases, the reference input signals are specified as set-points represented by computer code words and stored in computer memory which is already in a digital format. Thus the format of the feedback information and the reference input will be compatible and the tracking error $e(k)$ can be obtained quite readily with

digital computer arithmetic. Furthermore, when $r(k)$ is specified as a computer code word, the A/D converter need not be included explicitly for the reference input in the system. It is required only when $r(k)$ is an externally applied signal.

3.3 Controller

Before one can evaluate the performance of an ADPCM A/D converter in a control loop, a well-defined controller, or compensator, must be specified. Among the many controller design methods, both classical and modern, the Linear-Quadratic-Gaussian (LQG) method is a very suitable choice (Kwakernaak and Sivan, 1972; Moroney, 1983). It has many desirable features such as robustness to plant parameter variations, optimal performance, stability, and its multivariable nature (Kwakernaak and Sivan, 1972; Moroney, 1983). Furthermore, the controller parameters can be selected in a more straightforward and systematic manner than that of the other controller design methods such as PID controller design which may require careful tuning of the parameters by trial and error (Kuo, 1980). More specifically, an LQG control formulation has an explicit cost function from which a designer has a high degree of freedom in specifying the controller parameters in an optimal way. It can also provide a guideline for making trade-offs between the plant output error and the control efforts.

Consider a time-invariant SISO plant having a zero-order-hold preceding the plant input. It can be expressed by the discrete-time state-space equation

$$\mathbf{x}(k+1) = \Phi\mathbf{x}(k) + \Gamma u(k) + \Gamma_1 w(k) \tag{3.1}$$

$$\begin{aligned} z(k) &= \mathbf{H}\mathbf{x}(k) + v(k) \\ &= y(k) + v(k) \end{aligned} \tag{3.2}$$

where Φ $(n\times n)$, Γ $(n\times 1)$, Γ_1 $(n\times 1)$, and $\mathbf{H}$ $(1\times n)$ are constant plant transition matrices; $\mathbf{x}(k)$ is the n-dimensional plant state vector; $u(k)$ is the piecewise constant control signal held constant at each sampling time by the zero-order-hold; $w(k)$, the process disturbance, and $v(k)$, the measurement noise, are independent stationary, white, zero mean, Gaussian random noise sequences. It can be shown that the open-loop transfer function is given by

$$G(z) = \frac{Y(z)}{u(z)} \triangleq [\mathbf{H}(z\mathbf{I}-\Phi)^{-1}\Gamma]. \tag{3.3}$$

Given the state-space model (3.1) and the noisy plant output measurements (3.2), the objective of an LQG regulator $[r(k)=0]$ problem is to design a linear discrete-time controller which can generate an optimal control sequence $u(k)$ to minimize a quadratic cost functional (Franklin and Powell, 1980)

$$J = \frac{1}{2}\sum_{k=0}^{N} [\mathbf{x}^T(k)\mathbf{Q}\mathbf{x}(k) + qu^2(k)] \tag{3.4}$$

subject to (3.1) and (3.2). Q is a non-negative weighting matrix on the plant state vector and q is a non-negative scalar weighting factor on the control signal. Q and q are chosen by the designer according to the relative importance or costs of the plant state errors and the control effort (Franklin and Powell, 1980). The solution can be solved by applying a discrete-time Kalman filter to the noisy plant output measurement to generate optimum plant state estimations (Anderson and Moore, 1979; Franklin and Powell, 1980):

$$\hat{\mathbf{x}}(k|k) = \bar{\mathbf{x}}(k|k-1) + \mathbf{L}(k)[z(k) - \mathbf{H}\bar{\mathbf{x}}(k|k-1)], \quad (3.5)$$

$$\bar{\mathbf{x}}(k+1|k) = \Phi\hat{\mathbf{x}}(k|k) + \Gamma u(k), \quad (3.6)$$

where $\hat{\mathbf{x}}(k|k)$ is the Kalman filter plant state estimate and $\bar{\mathbf{x}}(k|k-1)$ is the one-step-ahead plant state prediction. $\mathbf{L}(k)$ is the time-varying Kalman filter gain vector. The optimal control sequence can then be generated by

$$u(k) = -\mathbf{K}(k)\hat{\mathbf{x}}(k|k), \quad (3.7)$$

where $\mathbf{K}(k)$ is the time-varying optimal state feedback control gain vector which will minimize (3.4). Both $\mathbf{K}(k)$ and $\mathbf{L}(k)$ require the solving of discrete-time matrix Riccati equations and can be computed off-line when the parameters of the Kalman filter and the optimal control equations are specified. To simplify the computation, one can use a steady-state solution for both the optimal state feedback

control gain and the Kalman filtering problem (Kwakernaak and Sivan, 1972; Franklin and Powell, 1980). The resulting gain vectors **K** and **L** will then be constant and time-invariant.

From (3.5), (3.6), and (3.7), it can be seen that the control $u(k)$ depends on $\hat{x}(k|k)$ which requires measurement at time k. In a real implementation, because of the time required for the computations of (3.5), (3.6), and (3.7), there is an inevitable delay from the time the measurement is made at time k to the time the control $u(k)$ is ready to be applied to the plant input. This problem can be solved by a procedure suggested by Kwakernaak and Sivan (1972) and Moroney (1983). For the system considered, it is assumed that this delay is negligible when compared with the sampling period and that the control $u(k)$ is essentially available at time k.

For a SISO plant, one may be interested only in the plant output error and the cost of the control effort. In such a situation, one may only need to consider a cost function or loss function of the form

$$\ell = \frac{1}{2}\sum_{k=0}^{N} [\rho y^2(k) + u^2(k)] \qquad (3.8)$$

where ρ is the LQG weighting factor and is simply a scalar equivalent of **Q** in (3.4). The designer can specify the cost function such that a large ρ implies relatively inexpensive control and a relatively important plant output error which

is heavily penalized. For a low ρ, it implies that the control is relatively expensive and the plant output error is relatively unimportant.

Since a reference input is also included in the control system of Fig. 3.1, one can include the reference input in formulating the LQG problem and penalize the tracking error in the loss function:

$$\begin{aligned} J &= \frac{1}{2}\sum_{k=0}^{N} \{\rho[r(k) - y(k)]^2 + u^2(k)\} \\ &= \frac{1}{2}\sum_{k=0}^{N} \{\rho e^2(k) + u^2(k)\}. \end{aligned} \tag{3.9}$$

This results in an equivalent LQG tracking error regulator problem. For the system in Fig. 3.1, the time-invariant steady-state LQG optimal compensator transfer function can be shown to be (Appendix A)

$$D(z) = \frac{z\mathbf{K}(z\mathbf{I}-\Phi+\mathbf{L}\mathbf{H}\Phi)^{-1}\mathbf{L}}{1+\mathbf{K}(z\mathbf{I}-\Phi+\mathbf{L}\mathbf{H}\Phi)^{-1}(\mathbf{I}-\mathbf{L}\mathbf{H})\Gamma}, \tag{3.10}$$

where **K** and **L** are the time-invariant steady-state optimal control gain and Kalman filter gain vectors, respectively, as defined before.

3.4 Predictor

When applying an ADPCM system to a control problem, one is in a much better position, when designing the ADPCM predictor, than when applying ADPCM to speech coding

problems. Unlike speech coding problems where the source statistics are nonstationary and the knowledge of its model is often inexact and inadequate, the source model of a control system, which is the controlled plant and the controller, can usually be described quite accurately by the well defined dynamic equations of the system. When this better information about the source model is utilized in the design of the predictor, one can expect better ADPCM performance in a control system than that in a speech coding system. Furthermore, unlike speech coding problems which may require complex adaptive predictors to account for the nonstationarity of the source statistics, adaptive predictors are not required as long as the controlled plant is assumed to be time-invariant.

When designing the ADPCM predictor, one obvious objective is to obtain the best prediction such that the prediction error $\delta(k)$ would be a minimum. This results in a reduction in the dynamic range of the signal to be quantized and hence the quantization error can also be reduced. As can be seen from Fig. 3.1, the predictor $P(z)$ uses a quantized version of the plant output to generate the prediction. One may consider applying an optimal estimator designed for quantized measurements such that the mean square prediction error will be minimized. Such an estimator is, however, not practical because its time-growing memory requirements make it non-implementable (Curry, 1970). Other suboptimal alternatives may need be considered to approximate the

optimal estimator.

If one considers $\delta_q(k)$, the quantized prediction error, as an innovation sequence with $\epsilon(k)$, the quantization error, regarded as the measurement noise, one can consider applying a linear Kalman filter as the predictor. It is well known that if both the plant disturbance $w(k)$ and the quantization error $\epsilon(k)$ are Gaussian random processes, a linear Kalman filter can yield optimal (least mean square error) prediction. One further advantage is that effects of the quantization error $\epsilon(k)$ can be reduced greatly by the Kalman filter whereas, in the transversal filter predictor in a speech coding problem, the quantization error effects cannot be suppressed (Gunn and Sage, 1973). However, the quantization error $\epsilon(k)$ is usually far from being Gaussian. As a result, the linear Kalman filter performance is only suboptimal. Nevertheless, it allows a simple and straightforward procedure for specifying and implementing the predictor rather than the intractable optimal predictor designed for quantized measurements (Curry, 1970; Rink, 1982). If a steady-state Kalman filter (Franklin and Powell, 1980) is assumed, the implementation of the predictor can further be simplified. Thus, to simplify the implementation, a steady-state linear Kalman filter will be assumed for the predictor in the system considered here.

Given the discrete-time state-space model (3.1) of the plant with noisy measurements

$$z(k) = y(k) + \epsilon(k), \tag{3.11}$$

where $\epsilon(k)$ is the quantization error which is treated as the measurement noise, the Kalman filter equations (3.5) and (3.6) can be modified to (Rink, 1982):

$$\hat{\mathbf{x}}(k|k) = \bar{\mathbf{x}}(k|k-1) + \mathbf{L}\delta_q(k), \tag{3.12}$$

$$\bar{\mathbf{x}}(k+1|k) = \Phi\hat{\mathbf{x}}(k|k) + \Gamma u(k), \tag{3.13}$$

where $\mathbf{L}$ is the steady-state Kalman filter gain vector. Since it is unimportant whether the measurement error $\epsilon(k)$ is introduced inside or before the filter loop, the quantized prediction error $\delta_q(k)$ is taken to be the Kalman filter innovation (Rink, 1982):

$$\begin{aligned}\delta_q(k) &= Q_\epsilon[\delta(k)] \\ &= Q_\epsilon[r(k) - \bar{y}(k|k-1)] \\ &\cong y(k) + \epsilon(k) - \bar{y}(k|k-1) \\ &= z(k) - \mathbf{H}\bar{\mathbf{x}}(k|k-1),\end{aligned} \tag{3.14}$$

where the one-step-ahead prediction is obtained from

$$\bar{y}(k|k-1) = \mathbf{H}\bar{\mathbf{x}}(k|k-1). \tag{3.15}$$

One should note that $\hat{y}(k|k) \neq \mathbf{H}\hat{\mathbf{x}}(k|k)$ but rather (cf. Eq. (2.4))

$$\hat{y}(k|k) = \mathbf{H}\bar{\mathbf{x}}(k|k-1) + \delta_q(k)$$
$$= y(k) + \epsilon(k) = z(k). \qquad (3.16)$$

Assume for the moment a deterministic problem so that one can write the z-transforms of Eqs. (3.12), (3.13), and (3.14) to solve for

$$\bar{y}(z) = \mathbf{H}\bar{\mathbf{x}}(z) = \mathbf{H}(z\mathbf{I}-\Phi)^{-1}[\Phi\mathbf{L}\delta_q(z) + \Gamma u(k)]. \qquad (3.17)$$

From the block diagram Fig. 3.1,

$$\bar{y}(z) = F(z)u(z) + P(z)[\bar{y}(z) + \delta_q(z)] \qquad (3.18)$$

and solving for $\bar{y}(z)$ gives

$$\bar{y}(z) = \frac{P(z)}{1 - P(z)}\delta_q(z) + \frac{F(z)}{1 - P(z)}u(z). \qquad (3.19)$$

Comparing (3.17) and (3.19), it follows immediately that

$$\frac{P(z)}{1 - P(z)} = \mathbf{H}(z\mathbf{I}-\Phi)^{-1}\Phi\mathbf{L} \qquad (3.20)$$

and

$$\frac{F(z)}{1 - P(z)} = \mathbf{H}(z\mathbf{I}-\Phi)^{-1}\Gamma \triangleq G(z). \qquad (3.21)$$

From Eqs. (3.20) and (3.21), the predictor and control feedforward transfer function can readily be obtained:

$$P(z) = \frac{\mathbf{H}(z\mathbf{I}-\Phi)^{-1}\Phi\mathbf{L}}{1 + \mathbf{H}(z\mathbf{I}-\Phi)^{-1}\Phi\mathbf{L}} \tag{3.22}$$

and

$$F(z) = \frac{G(z)}{1 + \mathbf{H}(z\mathbf{I}-\Phi)^{-1}\Phi\mathbf{L}}$$

$$= G(z)[1 - P(z)]. \tag{3.23}$$

It can be shown that both $P(z)$ and $F(z)$ contain closed-loop poles from the Kalman filter state estimator equation and hence they are designed to be stable. The design of $P(z)$ and $F(z)$ is not yet complete, however. The Kalman filter gain $\mathbf{L}$ in these transfer functions depends on the variances of the process disturbance w and the quantization noise ϵ, which remain to be specified, and is to be solved from the steady-state matrix Riccati equation for the Kalman filter (Franklin and Powell, 1980).

From (3.22) and (3.23), it can readily be seen that the design of the ADPCM predictor is completely independent of the controller. This property could be advantageous because it implies that when the controller need be changed, which may be necessary during the initial design of the control system, it is not necessary to redesign the predictor and the quantizer. A much simplified design procedure will thus result.

In Fig. 3.1, one may notice that the controller $D(z)$, the predictor $P(z)$, and the feedforward transfer function

$F(z)$ are represented as completely separate blocks for the sake of convenience. But in many cases, such as the LQG design presented here, they may involve some plant state estimations and hence they could share some common elements (Rink, 1982). In fact, for an LQG design, one may combine $D(z)$, $P(z)$, and $F(z)$ and design them altogether as a state feedback control system with state estimates generated by the Kalman filter, using the quantized innovation δ_q as the filter input.

3.5 Adaptive Quantizer

The adaptive quantizer to be used in the ADPCM quantizer of Fig. 3.1 is assumed to have an input-output characteristic, at time k, as shown in Fig. 3.2. It has a uniform quantizer characteristic in which the quantization level spacings are uniformly distributed. When the digital representation of the quantizer output is to be processed by a digital system, uniform quantization is preferred, rather than logarithmic companded or optimum nonuniform quantization which may require extra complex operations before the output can be processed by the digital system (Rabiner and Schafer, 1978). In most cases, the digital controller and predictor algorithms are generally realized and implemented with microprocessors, minicomputers, or special-purpose digital signal processing hardware in which two's complement arithmetic is most commonly used. Thus, if the quantizer output is coded with and represented by a

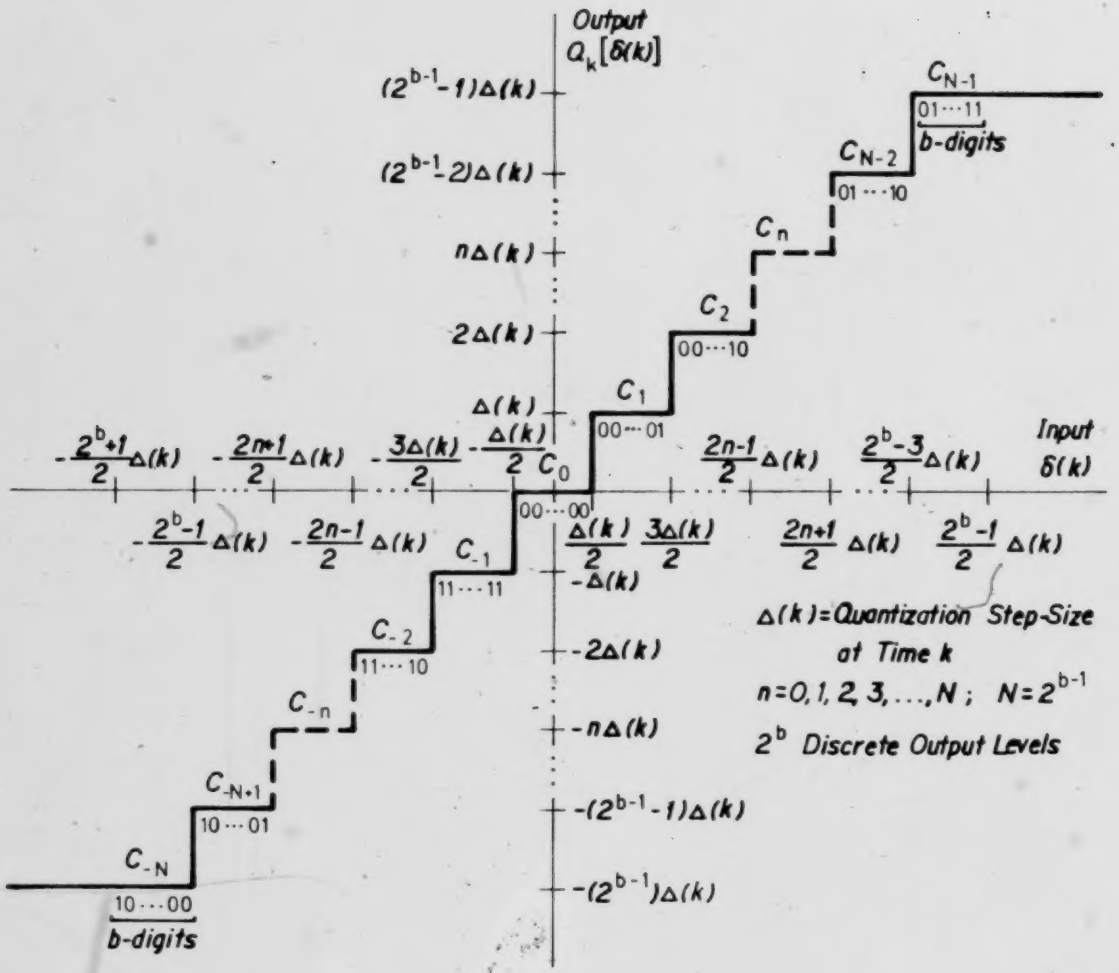


Fig. 3.2 Input-Output Characteristic of a b-Bit Adaptive Quantizer.

two's complement digital binary number, the digitized quantizer output can then be used directly in the computations in the digital control system. Furthermore, when using two's complement binary coding, the code words can be assigned in such a way that they can serve as a direct numerical representation of the magnitudes of the quantizer input sample values (Rabiner and Schafer, 1978). For this reason, a mid-tread quantizer characteristic, as shown in Fig. 3.2, in which a zero level output is available when the input samples are in the vicinity of zero, is more favourable than a mid-riser quantizer which cannot provide an unambiguous zero level output (Gersho, 1978; Rabiner and Schafer, 1978). One may notice that when the quantizer output is coded with a b-bit binary number, there can only be 2^b discrete output levels. For a mid-tread quantizer, this will lead to one more negative output level than positive (Rabiner and Schafer, 1978). However, this extra negative level, which can be regarded as the negative full scale level, is not used in the computations since it is defined in digital computing that the most negative (and the most positive) value is the output level which has a magnitude of the full scale minus one quantization step-size (Sheingold, 1977). For a b-bit quantizer, this magnitude corresponds to $(2^{b-1}-1)\Delta(k)$ (See Fig. 3.2). Nevertheless, the negative full scale code is useful for quantizer checking and as an adjustment code (Sheingold, 1977). An example of two's complement binary number representations

for the quantizer output levels is also shown in Fig. 3.2.

In the uniform quantizer of Fig. 3.2, it can be seen that the input decision levels and the output quantization levels are all equally spaced and are separated by one quantization step-size $\Delta(k)$ at time k. When the magnitude of the input sample value falls into the n-th input decision interval,

$$\frac{2n-1}{2}\Delta(k) \leq |\delta(k)| \leq \frac{2n+1}{2}\Delta(k), \tag{3.24}$$

it will then be represented by the n-th quantizer output level. For a b-bit quantizer, these 2 discrete output levels are given by

$$Q_x[|\delta(k)|] = n\Delta(k), \quad \text{for } n=0,1,2,\ldots,N;\ N=2^{b-1}; \tag{3.25}$$

which are coded with their corresponding digital code words C_n (for n=0, 1, 2, 3, ..., N). Since the negative full scale code C_{-N} is never used in the computation, practically, there are only 2^b-1 (or $2N-1$) quantizer output levels available. If the input sample value is assumed to be bounded as $|\delta(k)| \leq \delta_{max}(k)$, then the quantizer step-size is determined by

$$\Delta(k) = \frac{\delta_{max}(k)}{2^{b-1}}. \tag{3.26}$$

Equivalently, if given the quantization step-size $\Delta(k)$, the

maximum value that the quantizer can represent, or the full scale range $\delta_{FS}(k)$, is then determined by

$$\delta_{FS}(k) = 2^{b-1}\Delta(k). \tag{3.27}$$

It can readily be seen that a uniform quantizer can be specified completely if given b and $\Delta(k)$ or $\delta_{max}(k)$ (Rabiner and Schafer, 1978).

Since approximations are involved in the quantization process, an inherent error is unavoidable. Generally, this error is considered as an additive noise given by

$$\epsilon(k) = \delta_q(k) - \delta(k), \tag{3.28}$$

where $\epsilon(k)$ is the quantization error; $\delta(k)$ is the unquantized input sample, and $\delta_q(k)$ is the quantizer output value ($\delta_q(k)=Q_x[\delta(k)]$). When the input sample value is within the quantizer range, it is well known that the error $\epsilon(k)$ for a uniform quantizer is bounded by half the quantization step-size

$$|\epsilon(k)| \leq \frac{\Delta(k)}{2}, \tag{3.29}$$

and is regarded as "granular" noise (Gersho, 1978; Rabiner and Schafer, 1978). However, when the input sample value is outside the full scale quantizer range, the quantization error is no longer bounded and the quantizer is said to be "overloaded" (Gersho, 1978). A well designed quantizer

should be matched to the quantizer input probability density function to reduce the possibility of overloading the quantizer while maintaining an acceptable low granular noise level. This requires a compromise between accommodating a wide dynamic range and obtaining a low quantization step-size to reduce the quantization error (Goodman and Gersho, 1974).

In many cases, the quantizer input samples may cover a wide dynamic range and the signal variance of these samples may not be known. It may be difficult to select a fixed quantizer step-size that will provide the best compromise between the quantizer dynamic range and the granular noise for the input samples. If, however, the step-size is allowed to vary continually, according to a time-invariant adaptation scheme, to match the quantizer dynamic range to the input signal variance, then the quantizer will be able to handle input samples having large amplitude variations, especially nonstationary signals, and having unknown signal variance (Jayant, 1973). One should note, however, that the main advantage of adaptation over nonadaptive quantization strategy is the increase in the dynamic range rather than a gain in the inherent SNR (Jayant, 1973). With step-size adaptation, it is possible to maintain a specified SNR over a wider dynamic range than that of a nonadaptive one.

In order to match the quantizer to the input signal variance, it is desirable that the adaptation scheme be able to adjust the step-sizes to values that will result in

appropriate quantizer ranges for the changes in the current power level of the input samples (Goodman and Gersho, 1974). This will require some form of estimation for the input signal variance and thus requires the use of quantizer memory when the input samples are locally stationary (Jayant, 1974). Noll (1974, 1975a) described several approaches of exploiting the quantizer memory in the estimation of the input signal variance but they may require a large amount of computations and quantizer memory. Another adaptation scheme, proposed by Jayant (1973), which does not require an explicit estimate of the input signal varinace, is found to be much more efficient and simpler to implement (Jayant, 1973; Cummiskey *et al.*, 1973). This algorithm, which can be shown to be equivalent to an algorithm that forms a maximum likelihood estimate of the input signal variance (Cohn and Melsa, 1975), adapts the quantizer step-sizes by making instantaneous exponential changes based on information derived from either the quantizer output code-words or input sample values and requires only one word of quantizer memory (Jayant, 1973; Cummiskey *et al.*, 1973). In the adaptive quantizer used in Fig. 3.1, a one-word memory step-size adaptation algorithm will be considered.

3.5.1 Feedback Adaptation

The one-word memory feedback adaptation algorithm discussed here is similar to the one studied by Goldstein and Liu (1976, 1977a, b, c). It is a particular realization

of the systems described by Cummiskey *et al.* (1973) and Goodman and Gersho (1974) and is very simple and straightforward to implement. Particularly, if the parameters of the algorithm are chosen appropriately, it is very practical and suitable for microprocessor, minicomputer, or special-purpose digital signal processing hardware implementations.

In this algorithm, the only information required in making changes of step-sizes, after each new input sample, is the knowledge of the magnitude of the quantizer output level associated with the previous input sample (Jayant, 1973; Goodman and Gersho, 1974). The step-size $\Delta(k+1)$ at time $k+1$ is either increased or decreased by a multiplicative factor γ (usually $1 \le \gamma \le 2$) from its previous value at time k, depending on the magnitude of the quantizer output level at time k relative to a given threshold (Goldstein and Liu, 1976, 1977a, b, c). The algorithm can be summarized as follows

$$\Delta(k+1) = \begin{cases} \Delta_{max} = \gamma^{M}\Delta_0, \\ \quad \text{if } \gamma\Delta(k) \ge \Delta_{max} \text{ and } |Q_v[\delta(k)]| \ge 2^{b-2}C\Delta(k) \\ \gamma\Delta(k), \\ \quad \text{if } \gamma\Delta(k) < \Delta_{max} \text{ and } |Q_v[\delta(k)]| \ge 2^{b-2}C\Delta(k) \\ \Delta(k)/\gamma, \\ \quad \text{if } \Delta(k)/\gamma > \Delta_{min} \text{ and } |Q_v[\delta(k)]| < 2^{b-2}C\Delta(k) \\ \Delta_{min} = \gamma^{-M}\Delta_0, \\ \quad \text{if } \Delta(k)/\gamma \le \Delta_{min} \text{ and } |Q_v[\delta(k)]| < 2^{b-2}C\Delta(k) \end{cases} \tag{3.30}$$

where b is the number of quantizer bits, γ is the

multiplicative factor referred as the "step-size adaptation parameter", and C is the "adaptation threshold parameter" (Goldstein and Liu, 1976, 1977a, b, c). Δ_0 is a nominal or central step-size and M is a factor related to the dynamic range of the quantizer.

In this algorithm, only two distinct step-size adaptation multipliers are used. They are chosen in such a way that their product is one. This is motivated by the considerations of maintaining a well-defined set of step-sizes which are to be implemented with digital hardware (Goldstein and Liu, 1977a). Furthermore, they are selected such that when the quantizer output level is high, an indication of the quantizer range being too low for the current input sample, the step-size will be increased. On the other hand, when the output level is low, suggesting the range is too high for the current input, then the step-size will be decreased accordingly. Practical hardware implementations also require limitations on the allowable step-sizes. Thus, in the algorithm, the step-sizes are constrained to lie between the minimum value Δ_{min} and the maximum value Δ_{max}. These contraints will determine the dynamic range of the adaptive quantizer through the ratio $R_d=\Delta_{max}/\Delta_{min}=\gamma^{2M}$ (Jayant, 1974). This will be the range over which a specified SNR can be maintained by the adaptive quantizer.

When making step-size adjustments, the quantizer output is compared with a given adaptation threshold level C_T,

$$C_T = 2^{b-2}C, \tag{3.31}$$

which is determined by C and is relative to the half full scale output level (cf. (3.27)). If implemented with digital hardware, these output levels are represented by digital binary code-words. For a b-bit mid-tread quantizer, these output levels, given by $|Q_k[\delta(k)]|/\Delta(k)$, can only assume discrete values from the set $(1, 2, 3, \ldots, 2^{b-1}-1)$. To obtain an unambiguous decision threshold, C_T must be chosen from the uniquely decodable set $(1, 2, 3, \ldots, 2^{b-1}-1)$. This then limits C to (Goldstein and Liu, 1977a):

$$C \in \left(\frac{1}{2^{b-2}}, \frac{2}{2^{b-2}}, \frac{3}{2^{b-2}}, \ldots, 2-\frac{1}{2^{b-2}}\right). \tag{3.32}$$

The selection of C is crucial to the performance of the adaptive quantizer. When C is too small, C_T will be too low and a step-size larger than the optimum value will be used causing an increase in the granular noise. If C is too large, the quantizer will tend to adopt the smaller step-sizes and this will lead to frequent occurences of quantizer saturation† and slope overload‡ (Goldstein and Liu, 1977a). The best choice of C should result in a system where granular noise would be the dominant source of error (Goldstein and Liu, 1977a) since saturation and slope

---

†Quantizer saturation occurs when the input is larger than what the quantizer can handle. In other words, this happens when the quantizer is overloaded.

‡Slope overload occurs when the quantizer input signal is increasing at a rate faster than the adaptive quantizer can follow.

overload errors, which are unbounded, are more harmful than granularity error (Jayant, 1973).

3.5.2 Feedforward Adaptation

The one-word memory feedforward adaptation algorithm is basically the same as the feedback algorithm except that the magnitude of the quantizer input is used in the step-size adaptation. It can be summarized as follows (Goldstein and Liu, 1977b, c)

$$\Delta(k+1) = \begin{cases} \Delta_{max}=\gamma^{M}\Delta_0, \\ \quad \text{if } \gamma\Delta(k)\geq\Delta_{max} \text{ and } |\delta(k+1)|\geq 2^{b-2}C\Delta(k) \\ \gamma\Delta(k), \\ \quad \text{if } \gamma\Delta(k)<\Delta_{max} \text{ and } |\delta(k+1)|\geq 2^{b-2}C\Delta(k) \\ \Delta(k)/\gamma, \\ \quad \text{if } \Delta(k)/\gamma>\Delta_{min} \text{ and } |\delta(k+1)|<2^{b-2}C\Delta(k) \\ \Delta_{min}=\gamma^{-M}\Delta_0, \\ \quad \text{if } \Delta(k)/\gamma\leq\Delta_{min} \text{ and } |\delta(k+1)|<2^{b-2}C\Delta(k) \end{cases} \tag{3.33}$$

where b, C, and γ have the same significance as in the feedback algorithm. The advantage of the feedforward algorithm is that the most current input sample is used in the step-size adjustment whereas, in a feedback algorithm, information about the previous sample is used and so a one sample delay is inherent. The disadvantage is that a feedforward algorithm may be more complex to implement as compared with a feedback algorithm.

3.5.3 Selection of Adaptation Threshold Parameter

In both the feedback (3.30) and the feedforward (3.33) adaptation algorithms, the adaptation threshold parameter C is crucial to the quantizer performance, which in turn affects the performance of the control system. It should be selected such that the most suitable step-size value would be used by the quantizer for the current quantizer input power level.

In the system considered here, the quantizer input is the prediction error $\delta(k)$, of which the dynamic behaviour is undefined, for the moment, and hence its power level is also not available to the quantizer. Nevertheless, because of its adaptation ability, the quantizer will successively adjust its step-size in attempting to match to the unknown power level. From the adaptation algorithms, it can be seen that the step-size will be increased if $|Q_4[\delta(k)]| \geq C_T \Delta(k)$ or $|\delta(k)| \geq C_T \Delta(k)$ and decreased otherwise. This suggests that the step-size tends to drift towards a steady-state step-size value Δ_{ss} such that the quantizer will be scaled according to the long-term average square value $E[\delta^2(k)]$ and hence (Goldstein and Liu, 1977a):

$$E[\delta^2(k)] \simeq (C_T \Delta_{ss})^2. \qquad (3.34)$$

One should note, however, that with the algorithms presented here, a constant steady-state step-size can never be reached, even with a constant $\delta(k)$. Rather, the step-size will be centered at the steady-state value Δ_{ss} and increased

or decreased by the multiplicative factor γ at subsequent sampling times.

To select a suitable C parameter, consider a constant quantizer input $\delta(k)=\delta_o$ so that $E[\delta^2(k)]\rightarrow\delta_o^2$ in the steady-state. Given this δ_o value, one must choose C such that granular quantization noise will be the dominant source of error. Suppose steady-state has been reached and the quantizer step-size is centered at the steady-state value Δ_{ss} which is "optimum" for the given δ_o value. As mentioned above, with the adaptation algorithms (3.30) and (3.33), a fixed step-size cannot be reached. Thus the step-size can either be increased or decreased from Δ_{ss} by the multiplicative factor γ. If the step-size is decreased from Δ_{ss}, then the quantizer may not be adequate to quantize the given δ_o value, and this may result in possible quantizer saturation, which may introduce large errors. In order to prevent $\delta(k)$ from converging to smaller values, one can select a C value, or C_T, such that

$$\delta_o \geq C_T\Delta_{ss}. \qquad (3.35)$$

This implies that the input δ_o value should be at least as large as the threshold level so that the step-size will be switched up to $\gamma\Delta_{ss}$ at the next sampling period.

Now suppose the step-size has been increased from the steady-state value to $\gamma\Delta_{ss}$. If this keeps increasing at subsequent samples, then large quantization error will result as the step-size will no longer be suitable for the

given δ_0 value. In order to prevent it from diverging to larger values, one can select C (or C_T) such that

$$\delta_0 \leq (C_T - 1)\gamma\Delta_{ss}. \tag{3.36}$$

This implies that when the step-size is increased from the "optimum" steady-state value, the threshold level should be chosen in a way that will ensure the step-size to switch back to the steady-state value. In (3.36), this is done by setting the threshold, when $\Delta(k)=\gamma\Delta_{ss}$, to be at least one step-size higher than the given δ_0 value.

Combining (3.35) and (3.36), then

$$C_T \leq (C_T - 1)\gamma, \tag{3.37}$$

from which one can obtain

$$2^{b-2}C = C_T \geq \frac{\gamma}{\gamma-1} \tag{3.38}$$

and hence

$$C \geq \frac{4\gamma}{2^b(\gamma-1)}. \tag{3.39}$$

Eq. (3.39) thus provides a guideline for selecting the adaptation threshold parameter C. However, it only gives a lower limit on C. The best C value should be one that is large enough so that granular noise will be the dominant source of quantization error (Goldstein and Liu, 1977a).

Simulation results (Goldstein and Liu, 1977a) have shown that for $\gamma=2$ and $b=4$, $c=1/2$ would give best quantizer performance for a unit variance Gaussian quantizer input. From (3.39), with $\gamma=2$ and $b=4$, it is also found that $c \geq 1/2$.

Besides selecting the appropriate c parameter, one must also scale the quantizer (by choosing Δ_{max} and Δ_{min}) in such a way that the steady-state step-size value Δ_{ss} would be adequate to accommodate $\delta(k)$ with the probability of overloading the quantizer kept very low.

3.5.4 Quantizer Scaling

The scaling of the adaptive quantizer requires knowledge of the variance of the prediction error, $E[\delta^2(k)]$, which depends on the quantization error, plant disturbance, and the reference input. To obtain the relationship of $\delta(k)$ to these quantities, one can assume a deterministic problem, for the moment, and write down the transfer functions from ϵ, w, and r to δ. For the system configuration of Fig. 3.1, it can be shown that in general (Appendix B)

$$\delta(z) = G(z)w(z) - \frac{P(z)}{1 - P(z)}\delta_q(z). \qquad (3.40)$$

With suitable quantizer scaling and the appropriate choice of c so that quantizer overload seldom occurs, it can be assumed that granular noise is the dominant source of the quantization error. When this error is modelled as an additive noise, (3.40) can be rewritten as (Appendix B)

$$\delta(z) = F(z)w(z) - P(z)\epsilon(z). \qquad (3.41)$$

As can be seen from (3.40) and (3.41) the control u and the reference input r have no effects on the information $\delta(k)$ which is to be quantized. Since δ is free from the influence of the control and reference signals, it implies that the design of the quantizer, like the ADPCM predictor, can be completely separate from the design of the controller. Thus, unlike a conventional fixed quantizer, the quantizer in a differential quantization system need not be changed when a change in controller strategy arises.

Eq. (3.41) also suggests, when δ is within the quantizer range and granular noise assumption holds, that δ is simply a sum of filtered versions of the process disturbance w and the quantization noise ϵ. Relaxing the deterministic assumption and assuming w and ϵ to be white and zero mean stochastic processes which are mutually uncorrelated, the mean square prediction error can be expressed as

$$E[\delta^2] = \frac{1}{2\pi j}\oint_{|z|=1} |F(z)|^2 \frac{dz}{z} E[w^2] + \frac{1}{2\pi j}\oint_{|z|=1} |P(z)|^2 \frac{dz}{z} E[\epsilon^2]$$

$$= G_F \cdot R_W + G_P \cdot R_\epsilon, \qquad (3.42)$$

where it has been defined that

$$G_F = \frac{1}{2\pi j} \oint_{|z|=1} |F(z)|^2 \frac{dz}{z}; \quad R_w = E[w^2]; \tag{3.43}$$

and

$$G_p = \frac{1}{2\pi j} \oint_{|z|=1} |G(z)|^2 \frac{dz}{z}; \quad R_\epsilon = E[\epsilon^2]. \tag{3.44}$$

When the input signal is within the quantizer range, the quantization error is bounded by (2.3) or (3.29). The variance of the quantizer noise will then be given by

$$R_\epsilon = \frac{\Delta^2}{12} \tag{3.45}$$

for uniform quantization and a zero mean $\epsilon(k)$. The step-size Δ as given by (2.3) depends on the quantizer scaling factor used and hence depends on the quantizer loading. Defining

$$R \triangleq \frac{R_w}{R_\epsilon} \tag{3.46}$$

then (3.42) can be rewritten as

$$\frac{E[\delta^2]}{R_\epsilon} = G_F \cdot R + G_p. \tag{3.47}$$

During the design stage, one can assume a fixed quantizer scaling factor and hence a fixed step-size Δ. With a fixed step-size, the analysis problem simply reduces to that of a non-adaptive DPCM application in control similar

to that of Rink (1982). Once a scaling factor for a given R_w value is found, the adaptive quantizer step-sizes can then be found as discussed later. After solving (3.47) for R, together with (3.45) and (3.46) and a given R_w, one can then specify the quantizer scaling factor by solving for Δ which is related to R_ϵ as indicated in (3.45) above. The resulting Δ can then be used in determining Δ_{max} and Δ_{min} used in the adaptation algorithms (3.30) and (3.33).

To avoid quantizer overloads, one would choose a quantizer scaling factor such that the quantizer input is within the quantizer full scale range and the probability of quantizer overload would be kept very low. For example, if one approximates $\delta(k)$ as a zero mean Gaussian stochastic process, a conservative scaling factor is that of assigning δ_{max} and δ_{min} to be the three-sigma (3σ) values (Rink, 1982) so that the probability of overflow is about 2×10^{-3}. An even more conservative scaling is that of four-sigma (4σ) loading, a value commonly used in speech coding (Gersho, 1978), which will lead to a probability of overflow of about 6×10^{-5}.

For three-sigma loading $\delta^2_{max}=9E[\delta^2(k)]$ and $\Delta^2=36E[\delta^2(k)]/2^{2b}$. This results in

$$R_\epsilon = \frac{3\cdot E[\delta^2]}{2^{2b}}. \qquad (3.48)$$

For four-sigma loading, $\delta^2_{max}=16E[\delta^2(k)]$ and $\Delta^2=64E[\delta^2(k)]/2^{2b}$. This will give

$$R_{\epsilon} = \frac{16}{3} \cdot \frac{E[\delta^2]}{2^{2b}}. \qquad (3.49)$$

Putting (3.48) and (3.49) into (3.47), for 3σ loading, one would then obtain

$$\frac{2^{2b}}{3} = G_F \cdot R + G_p, \qquad (3.50)$$

and for 4σ loading,

$$\frac{3}{16} \cdot 2^{2b} = G_F \cdot R + G_p. \qquad (3.51)$$

In general, (3.50) and (3.51) can be expressed as

$$a \cdot 2^{2b} = G_F \cdot R + G_p, \qquad (3.52)$$

where a is a factor which depends on the quantizer loading factor.

When the plant model, the quantizer loading factor, and the number of quantizer bits, b, are all specified (3.52) can then be solved for R. An immediate difficulty arises, however. Eq. (3.52) cannot be solved in closed form since both G_F and G_p involve solutions to the steady-state Kalman filter equations which in turn depend on R (Kwakernaak and Sivan, 1972). Iterative numerical or graphical solutions (Rink, 1982) may be required. Examples will be given later to illustrate the graphical solution of (3.52).

Suppose (3.52) has been solved and the solution is given by R_s. From (3.45) and (3.46), for a given R_w, the step-size can readily be solved:

$$\Delta = \sqrt{12 \cdot \frac{R_w}{R_s}} \tag{3.53}$$

which can be used in specifying Δ_{max} and Δ_{min} in the adaptation algorithms as discussed later.

The limitation of a fixed quantizer is that the design is for a given fixed R_w value only. When R_w changes, it implies that Δ must be re-adjusted from (3.53) for the new R_w value and a change in the hardware may be necessary. In many cases, once the controller and quantizer have been implemented, it may not be possible to re-adjust the hardware. Moreover, in many real applications, w is not necessarily stationary and its variance is often not known precisely. In the steady-state Kalman filter equations, w is usually assumed to be stationary with a given fixed variance R_w. This is only a simplification of the design process because if a nonstationary w is assumed a time-varying Kalman filter may be needed, and this will result in a more involved design procedure. The ability of the adaptive quantizer to adapt itself to different environments can be of advantage in resolving the difficulty of scaling the quantizer when R_w is changed. By allowing the quantizer to adapt its step-sizes the quantizer can seek the most appropriate quantizer scaling factor for a given R_w value.

The range of R_W for which the quantizer can adapt is specified by the designer, depending on the performance requirements. For example, one may specify an expected worst case process disturbance $R_{W_{max}}$ and from (3.53)

$$\Delta_{max} = \sqrt{12 \cdot \frac{R_{W_{max}}}{R_s}} . \qquad (3.54)$$

The minimum step-size can then be determined from $R_d=\Delta_{max}/\Delta_{min}=\gamma^{2M}$. Another way of selecting Δ_{max} and Δ_{min} is by specifying a given central R_{W_0} value and using (3.53) to determine a central step-size Δ_0:

$$\Delta_0 = \sqrt{12 \cdot \frac{R_{W_0}}{R_s}} . \qquad (3.55)$$

Then Δ_{max} and Δ_{min} can be determined by $\Delta_{max}=\gamma^M\Delta_0$ and $\Delta_{min}=\gamma^{-M}\Delta_0$ (see Eqs. (3.30) and (3.33)). From (3.54) and (3.55) it can be seen that the range of R_W which can be handled by the adaptive quantizer can be selected by choosing γ and M appropriately. A higher M will imply a wider range whereas a lower M will result in a narrower range of R_W.

One further advantage of the application of an adaptive quantizer is that the time-invariant steady-state Kalman filter would be useful over a wider range of R_W, determined by γ and M, than that of a fixed quantizer. When R_W is changed, it is necessary to re-adjust the scaling of the quantizer from (3.53), if it is possible. Otherwise, the

ratio R_s cannot be maintained and this implies the Kalman filter would not be optimal for the new R_w value. With an adaptive quantizer, however, it is possible to maintain the ratio R_s over a much wider range than that of the fixed quantizer. As a result the Kalman filter would be optimal for a much wider range as specified by γ and M. This aspect is very useful in many real applications where it may not be possible to change the hardware once it is implemented. With an adaptive quantizer, a time-varying Kalman filter is not necessary to account for a time-varying R_w.

It should be pointed out that the design for $P(z)$ and $F(z)$ and the adaptive quantizer scaling is only a simplified approximate method. In fact, this is simply the design for a nonadaptive DPCM coding application in control. One should note that with the adaptive quantizer in the control loop, it is inadequate to assume a white quantization error $\epsilon(k)$. Because of the adaptation algorithms, this error sequence is signal dependent as well as time-varying. Not only is it correlated to w and δ, but it is also a nonwhite sequence. As a result the z-domain analysis is grossly inadequate. Nevertheless, it greatly simplifies the design process, resulting in a suboptimal system performance which may be adequate for many applications.

Chapter 4

SUGGESTED HARDWARE REALIZATIONS AND IMPLEMENTATION CONSIDERATIONS

4.1 Introduction

Having been properly designed and selected, the digital controller, the control feedforward transfer function, the ADPCM predictor, the ADPCM predictor feedback loop, and the adaptive quantizer must eventually be realized and implemented with practical digital hardware components such as microprocessors or special-purpose digital signal processing hardware. However, since an adaptive quantizer having time-varying step-sizes is present in the control loop, the implementation of these tranfer functions will be different from that of systems using fixed quantizers which have only fixed quantization step-sizes and thus, fixed scale factors. One must consider the time-varying scale factors associated with the adaptive quantizer and realize the transfer functions such that the internal scaling of the data in the digital computations would be consistent. Furthermore, the digital-to-analog (D/A) converters should also be scaled accordingly to provide consistently scaled analog output values. Goldstein and Liu (1976) have suggested a method of implementing nonrecursive digital filters for ADPCM coded signals. Their results cannot, however, be applied directly in the scaling of the controller transfer functions since they are usually

recursive in control system applications. Furthermore, the predictor used in Goldstein and Liu (1976) is a simple unit delay (a first-order predictor), which greatly simplifies the scale-factor problem, whereas in control systems the predictor order is often greater than unity. In the following, possible hardware realizations of an adaptive quantizer and the scaling of the D/A converters as well as the internal scaling considerations of transfer function realizations will be briefly described and discussed.

4.2 Adaptive Quantizer Realization

As mentioned before, the objective of an adaptive quantizer is to match the quantizer to the dynamic range of the quantizer input by allowing a set of possible quantization step-sizes, selected according to a time-invariant adaptation algorithm. This implies that an adaptive quantizer is composed of two basic elements: a quantizer with a time-varying full-scale range and a step-size adaptation control algorithm. Instead of considering a quantizer having a varying full-scale range, one can view an adaptive quantizer as being a fixed quantizer preceded by a time-varying gain or scaling factor. This leads to an alternative configuration for the adaptive quantizer as shown in Fig. 4.1 (Goodman and Gersho, 1974; Rabiner and Schafer, 1978; Johnston and Goodman, 1978).

In this configuration, the quantizer input at time k is first pre-scaled by a gain factor which is proportional to

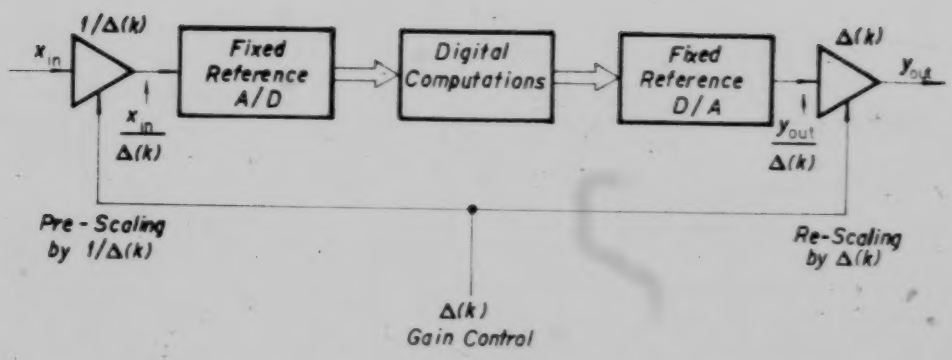


Fig. 4.1 Alternative Configuration of an Adaptive Quantizer [After Johnston and Goodman (1978)].

the inverse of the step-size $1/\Delta(k)$. It is then quantized with a fixed reference A/D converter. The digital output is then used in the digital computations. The end result is finally converted back to an analog signal and rescaled by a gain factor proportional to the the step-size $\Delta(k)$. This combination of a fixed quantizer with scaling and re-scaling is equivalent to adjusting the full-scale range of the quantizer and thus is equivalent to an adaptive quantizer having time-varying quantization step-sizes.

Different approaches of hardware implementation of an adaptive quantizer using the combination of a fixed quantizer with scaling and re-scaling have been reported recently (Johnston and Goodman, 1978; Boddie *et al.*, 1981). Johnston and Goodman (1978) have designed and constructed a multipurpose digital coding device for audio signals which is capable of providing ADPCM coding. The device is, however, based primarily on analog hardware components which may lead to problems of repeatability of the analog devices

(Boddie *et al.*, 1981). Another approach suggested by Boddie *et al.* (1981) is to use a special-purpose digital signal processor to configure the ADPCM coder and decoder. The entire process, being a digital operation, requires the quantizer input to be A/D converted into an 8-bit μ-law companded PCM format which is then converted into a 20-bit linear PCM code-word by the digital processor before the ADPCM coding procedure. This approach, though appropriate for communication applications where, in practice, the signals are usually represented digitally by a μ-law companded PCM format, is not applicable in the present application of an ADPCM coder as an A/D converter in a control loop. Not only does it require a redundant pre-quantization process with a μ-law converter, but also a complicated procedure for converting a shorter-wordlength log PCM code-word into a longer-wordlength linear PCM format which in many cases may not be permissible due to finite wordlength of the processor. Thus an alternative approach which is repeatable and reliable should be considered.

One attractive approach of realizing the adaptive quantizer scaling and re-scaling gain factors is to make use of a variable reference D/A converter (Sheingold, 1977). The Analog Devices AD7524 (Analog Devices, 1982), which is a multiplying D/A converter (MDAC), is particularly effective and useful for realizing a variable reference D/A converter. It can take in a digital code-word $D(k)$ and produce an analog output signal which is the product of $D(k)$ and the

reference input signal applied to the MDAC. If the step-size $\Delta(k)$ is represented by $D(k)$, the quantizer scaling factor is then realized by preceding the fixed quantizer by a variable reference D/A converter or a MDAC, together with an appropriately configured high gain operational amplifier (OP AMP), to produce an output signal which is proportional to the inverse of the step-size $1/\Delta(k)$ as shown in Fig. 4.2.

The re-scaling of the D/A output can also be realized by a similar procedure as shown in Fig. 4.3. Here, the desired output is first converted from its digital representation into an analog form by a fixed reference D/A converter with an appropriate reference voltage input. The output of the converter is then applied to the reference input of a variable reference D/A converter (or a MDAC). This second converter then acts as the re-scaling gain factor by taking in a digital input code-word $D(k)$ representing the step-size $\Delta(k)$ and produces the final desired re-scaled analog output signal.

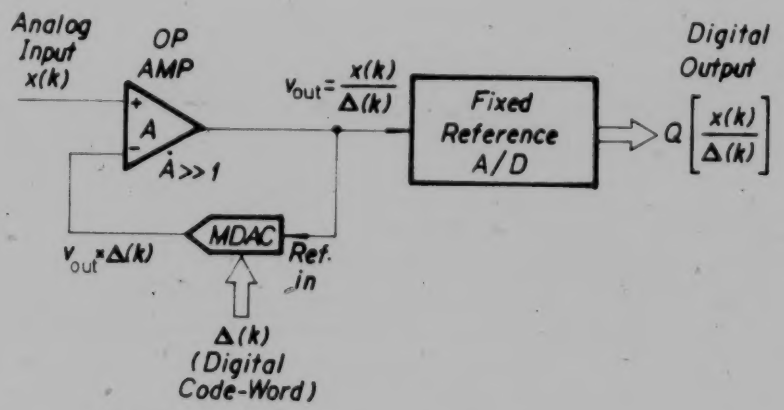


Fig. 4.2 Realization of Adaptive Quantizer Pre-Scaling.

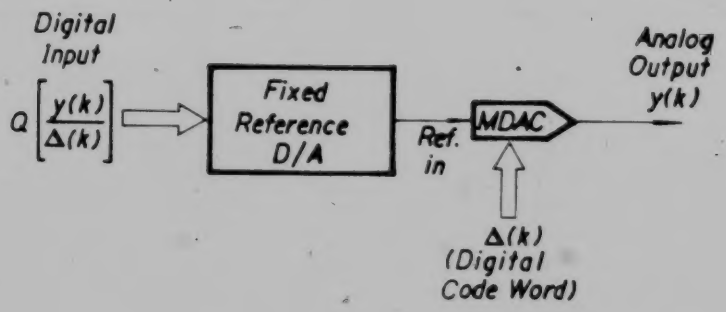


Fig. 4.3 Realizaiton of Analog Output Re-Scaling.

4.3 Adaptive Step-Size Control

The time-varying step-sizes used in the quantizer scaling can be updated with either the one-word memory feedback algorithm (3.30) or the one-word memory feedforward algorithm (3.33). The new step-size $\Delta(k+1)$ at time $k+1$ is obtained by either increasing or decreasing the past step-size value $\Delta(k)$ by a multiplicative factor γ. If this parameter is chosen appropriately, the resulting algorithm implementation can greatly be simplified. For example, by setting $\gamma=2$ and $\gamma^{-1}=1/2$, the multiplications of the step-sizes reduce simply to a one-bit left shift and a one-bit right shift, respectively, of the registers containing the past step-size value $\Delta(k)$ in sign magnitude convention. Not only can it simplify the adaptation operation, but it can also reduce the time for computing the new step-size value $\Delta(k+1)$ as well as the memory storage requirement for the step-sizes since only one word of memory location is all that is required to keep track of the

step-sizes. This approach is unlike the other methods of implementation (Cummiskey *et al.*, 1973; Johnston and Goodman, 1978; Boddie *et al.*, 1981) which may require more memory storage for a step-size look-up table. The price paid for this approach of simplified implementation by letting the multipliers to be $\gamma=2$ and $\gamma^{-1}=1/2$ is that the adaptive quantizer can only yield suboptimum performance because this set of multipliers is not optimal. Nevertheless, the ease of implementation as well as memory space savings, coupled with reasons related to the scaling problem explained later, make this approach seem more favourable than the others. If optimal performance is required one can use the optimal multipliers tabulated by Jayant (1973), but the implementation and the internal scaling problem will be more involved and complex.

For practical hardware implementations, the allowable step-sizes must be limited. In the algorithms (3.30) and (3.33), for $\gamma=2$, these are constrained to be $\Delta_{max}=2^{M}\Delta_0$ and $\Delta_{min}=2^{-M}\Delta_0$ where Δ_0 represents a nominal or central step-size value. The parameter M may be viewed as an integer related to the number of bits used for the exponent if the adaptive quuantizer is regarded as a particular form of a floating-point A/D converter (Oppenheim, 1978). It also constrains the number of allowable step-sizes to $2M+1$ and determines the step-size adaptation range as well as the dynamic range of the adaptive quantizer through the ratio $R_d=\Delta_{max}/\Delta_{min}$. This dynamic range is the range of input power

which can be handled by the quantizer and is the range over which a specified minimum SNR can be maintained (Jayant, 1974). Besides determining the dynamic range of the quantizer, M also determines the range of the exponent through the ratio

$$R_d = \frac{\Delta_{max}}{\Delta_{min}} = 2^{2M}. \tag{4.1}$$

The nominal step-size Δ_0 can alternatively be determined by

$$\Delta_0 = [\Delta_{max} \cdot \Delta_{min}]^{1/2} \tag{4.2}$$

if given Δ_{max} and Δ_{min} values which will meet certain requirements for R_d. If floating-point number representation is employed with m-bits for the exponent, which is related to the step-size used at time k, then from (4.1) the range of exponents is confined to be

$$2^m = 2M. \tag{4.3}$$

Thus the values of M are constrained to be

$$M = 2^{m-1}. \tag{4.4}$$

For example, with m=3 bits, then M=4 and the attainable dynamic range is R_d=256 or equivalently a dynamic range of 48 dB, which is quite adequate for many applications. Once the range of the step-sizes has been determined, the reference for the fixed A/D and D/A converters can be fixed

at a level such that the step-size of the fixed converters will correspond to the Δ_{max} value.

When all the parameters of the adaptation algorithms have been specified, the implementations of the algorithms can then be realized accordingly.

4.3.1 Feedback Algorithm

The implementation of the feedback adaptation algorithm is very simple and straightforward. Besides the analog components required for the quantizer pre-scaling, no extra analog components are required for this algorithm which can be realized with the computational power of the processor and its peripheral devices, such as memory components and interfacing devices, used in implementing the feedback control loop. One possible configuration is shown schematically in Fig. 4.4. When updating the step-size, information about the quantizer input signal is derived from the quantizer output level which is the quantized version of the input signal sample. This information is then used in the feedback adaptation algorithm (3.30). The step-size adaptation according to (3.30) can simply be obtained by comparing the absolute value of the b-bit digital output code-word from the A/D converter with the threshold code-word C_t associated with the threshold value $2^{b-2}C$ of (3.31) to determine whether the step-size should be increased or decreased. For a negative A/D output code-word represented by two's complement arithmetic, its absolute

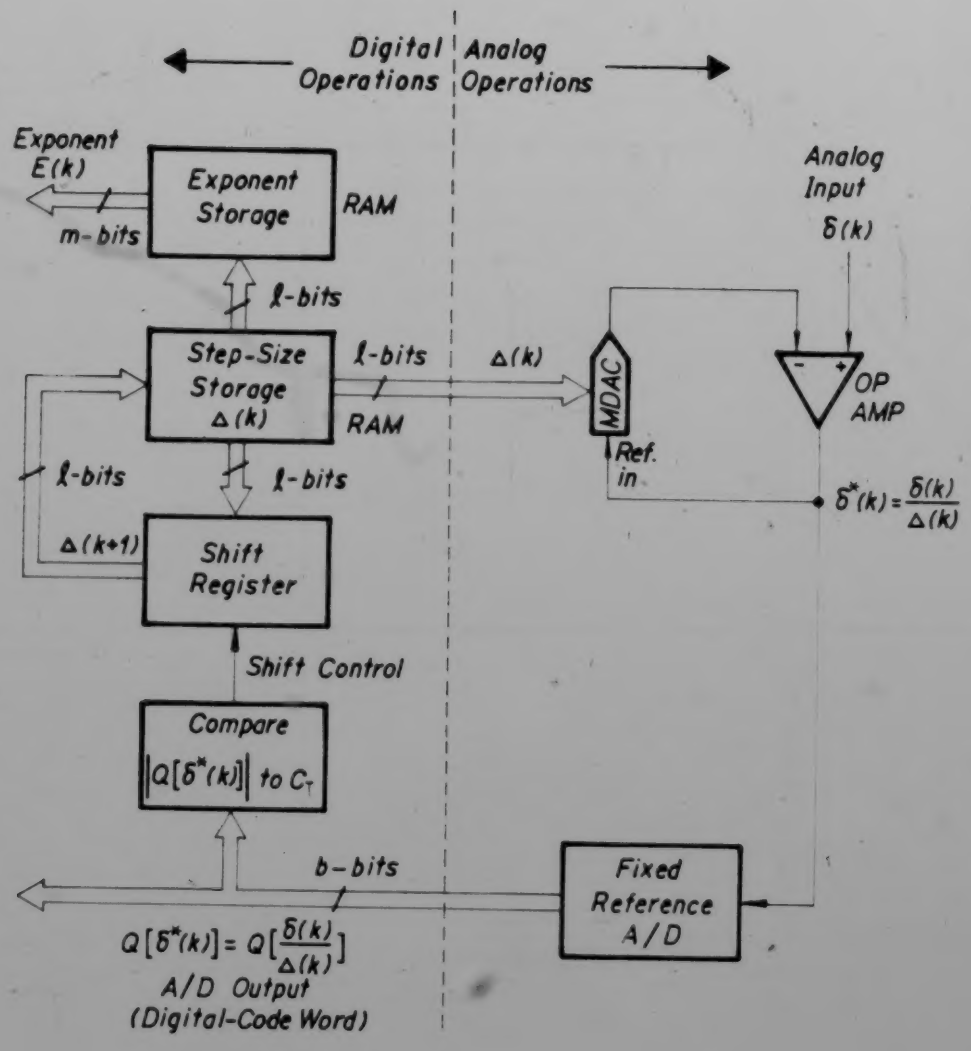


Fig. 4.4 Feedback Step-Size Adaptation Implementation.

value can be obtained by two's complementing the code-word to obtain its positive equivalent. The result of the comparison then controls the direction of shift for the ℓ-bit shift register, 8-bits in many cases, which contains the current step-size value $\Delta(k)$ at time k. After the shift operation, the content of the shift register then becomes the step-size value $\Delta(k+1)$ to be applied to the pre-scaling MDAC at the next sampling time $k+1$ and is stored into memory as well.

4.3.2 Feedforward Algorithm

One possible implementation of the feedforward algorithm is shown schematically in Fig. 4.5. In contrast to the feedback algorithm discussed previously, this algorithm requires extra analog hardware components in the implementation. Here, the current step-size $\Delta(k)$ at time k is updated based on information derived from the current quantizer input at time k. Since this information is available only in an analog form rather than in a digital form, comparing it with the threshold value requires some form of analog operations.

The threshold value in the feedforward algorithm is given by $2^{b-2}C\Delta(k-1)$. To compare it with the quantizer input which is in an analog form, it must be converted from its digital form into a compatible analog form. Therefore, the threshold value is converted from the digital code-word value $2^{b-2}C$ into an analog value using a fixed reference D/A

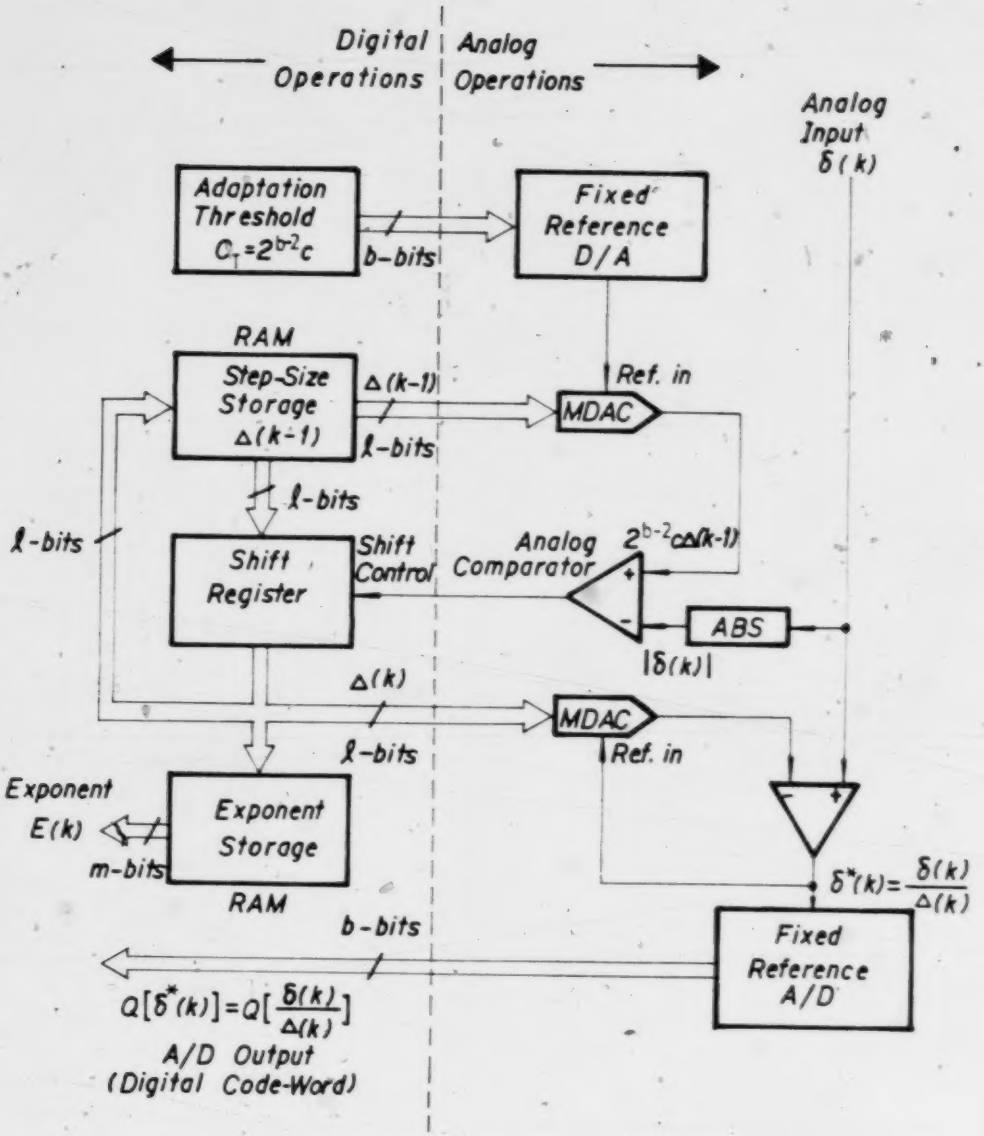


Fig. 4.5 Feedforward Step-Size Adaptation Implementation.

converter. It is then re-scaled by the past step-size value $\Delta(k-1)$ using a MDAC as shown in Fig. 4.5. The resulting analog version of the adaptation threshold is applied to the input of an analog comparator and is compared with the absolute value of the quantizer input obtained by passing it through an analog absolute value circuit. The comparator output is used to control the direction of shift for the ℓ-bit shift register which contains the past step-size value $\Delta(k-1)$. After the shift, the register then contains the current step-size value $\Delta(k)$ which is then used in the quantizer pre-scaling at time k. It is also stored into memory to be used in the step-size adaptation at the next sampling time.

As can be seen from Fig. 4.5, a more complex circuitry is necessary to implement the feedforward algorithm. Intuitively, this increased complexity may result in a better quantizer performance since the most current quantizer input value, and hence the most up-to-date information, is used to update the current step-size value whereas in a feedback algorithm, there is a one sample delay of the step-size value. The choice of either a feedback or a feedforward algorithm depends on the performance requirement as well as the circuit complexity constraints.

4.4 Internal Scaling of Data

When an adaptive quantizer is used in a digital feedback control loop, besides maintaining a consistent scaling between the A/D converter input and the D/A converter output, it is also necessary to provide a consistent scaling for all the internal data used in the controller, the control feedforward, and the ADPCM predictor transfer function computations as well as those used in the ADPCM prediction feedback loop. Fortunately, this internal scaling problem can be much simplified and is readily solvable if the transfer functions and the feedback loop are all realized with the concept of block-floating-point arithmetic (Oppenheim, 1970; Oppenheim and Schafer, 1975; Rabiner and Gold, 1975).

In block-floating-point number representation, a block or an array of numbers are jointly normalized with respect to a common exponent which is associated with the largest number in the block, represented as an ordinary floating-point number having a normalized mantissa. Unlike floating-point number representation which normalizes each number individually, block-floating-point only requires a single exponent for a large block of numbers and thus saves a lot of memory space (Rabiner and Gold, 1975).

When a digital filter is realized in block-floating-point arithmetic, the common exponent is associated with the largest number among the filter input and the intermediate filter states (i.e., the outputs of the

delay registers). These numbers are jointly normalized with respect to this common exponent and the computations (i.e., multiplications and additions) are carried out in fixed-point arithmetic. The scale factor associated with the common exponent in the normalization is then used to denormalize the filter output to produce a fixed-point result (Oppenheim, 1970).

Besides saving memory space, block-floating-point number representation, being a hybrid between fixed- and floating-point numbers, combines the advantages of floating-point number for its wide dynamic range and of fixed-point arithmetic for its simplicity and accuracy (Oppenheim, 1970).

If one considers an adaptive quantizer as a particular realization of a block-floating-point A/D converter, then the concept of block-floating-point arithmetic is readily applicable to the internal re-scaling problem of the feedback control loop having an ADPCM quantizer. In the following, this concept is applied to the overall scaling of the control system of Fig. 3.1 as well as the scaling problem of the transfer functions.

4.4.1 Transfer Function Realizations

In most cases, the digital controller or the predictor transfer functions are realized in a recursive form. For reasons of quantization effect reduction, they are usually implemented by cascading first- and second-order transfer

function sections to form higher order transfer functions (Oppenheim and Schafer, 1975; Rabiner and Gold, 1975). Therefore, considerations of the scaling problem for first- and second-order transfer functions will suffice.

To realize the transfer functions with block-floating-point arithmetic, one may regard the adaptive quantizer as a special case of a block-floating-point A/D converter. At each sampling time k, all the data in the feedback control loop can be considered as having been jointly normalized with respect to the current quantizer step-size $\Delta(k)$. The computations are then carried out in fixed-point arithmetic. The end result is then denormalized with the same scale factor $\Delta(k)$ to produce a consistently scaled fixed-point output. In this way, the step-size $\Delta(k)$ can be treated as an equivalent to the common exponent in a block-floating-point number representation.

4.4.1.1 First-Order Transfer Function

Consider a first-order transfer function in a recursive form

$$\frac{Y(z)}{X(z)} = \frac{a_0 + a_1 z^{-1}}{1 + b_1 z^{-1}}. \qquad (4.5)$$

The corresponding difference equation in the time-domain is

$$y(k) = a_0 x(k) + a_1 x(k-1) - b_1 y(k-1). \qquad (4.6)$$

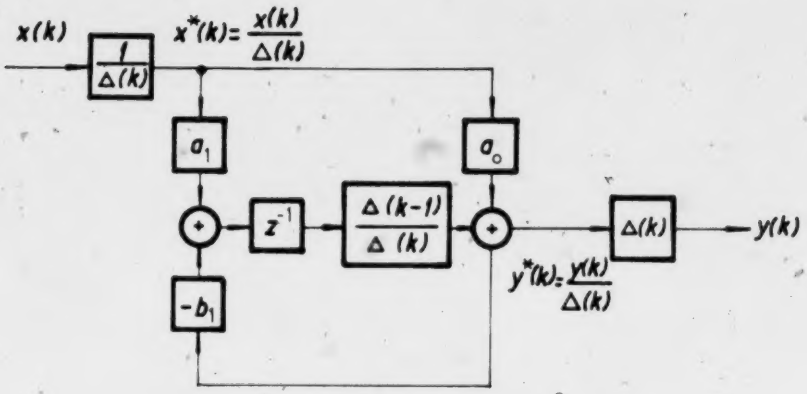
Normalizing it with respect to $\Delta(k)$ will result in

$$y^*(k) = a_0 x^*(k) + [a_1 x^*(k-1) - b_1 y^*(k-1)] \frac{\Delta(k-1)}{\Delta(k)}, \tag{4.7}$$

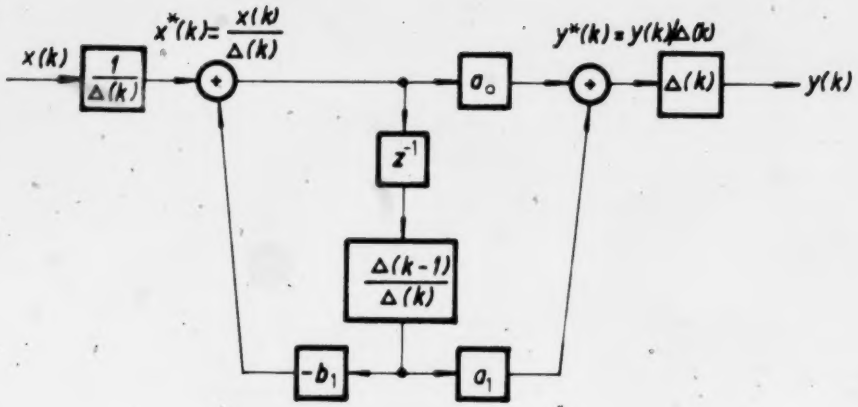
where

$$y^*(k) = \frac{y(k)}{\Delta(k)} \text{ and } x^*(k) = \frac{x(k)}{\Delta(k)}. \tag{4.8}$$

When realizing this first-order transfer function, Eq. (4.7) simply suggests that the outputs of the delay registers be rescaled by a scale factor of $\Delta(k-1)/\Delta(k)$ (Oppenheim, 1970). Fig. 4.6 shows the scale factor incorporated into the transfer function in a direct observer canonical form (Fig. 4.6a) and a direct control canonical form (Fig. 4.6b) (Franklin and Powell, 1980). Basically, the scale factor can be considered as consisting of a denormalizing and renormalizing process. Since the input to the transfer function is normalized with respect to $\Delta(k)$ at time k, the outputs at the delay registers represent numbers which are normalized with respect to the delayed version of the past step-size scale factor $\Delta(k-1)$. Thus, to obtain consistently scaled data, it is necessary to first denormalize them with $\Delta(k-1)$ and then renormalize them with $\Delta(k)$. In this way, the data involved in all the summations in the transfer function at time k would be jointly normalized with respect to $\Delta(k)$ and the scaling would then be consistent. The final output is then denormalized with $\Delta(k)$ to produce the fixed-point result as shown in Fig. 4.6.



(a) Direct Observer Canonical Form



(b) Direct Control Canonical Form

Fig. 4.6 Realization of a First-Order Transfer Function with Consistent Internal Scaling.

4.4.1.2 Second-Order Transfer Function

The realization of a second-order transfer function is identical to that of the first-order transfer function. Consider a second-order transfer function in a recursive form

$$\frac{Y(z)}{X(z)} = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2}}{1 + b_1 z^{-1} + b_2 z^{-2}}. \tag{4.9}$$

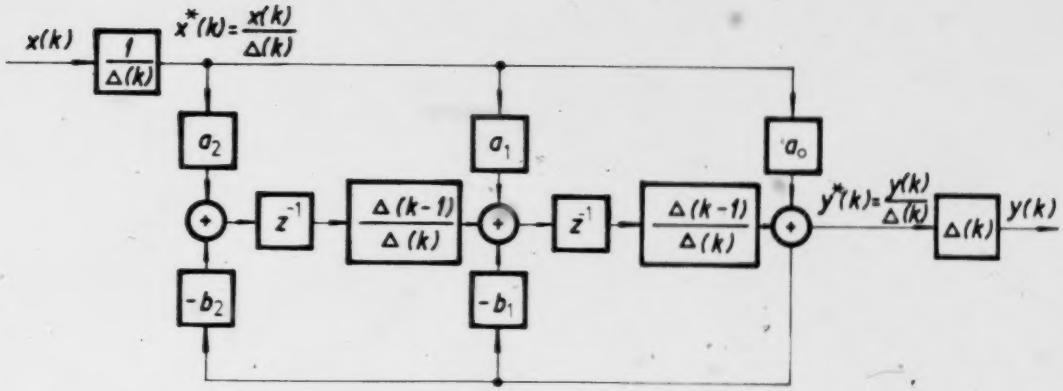
The corresponding difference equation in the time-domain is

$$y(k) = a_0 x(k) + a_1 x(k-1) + a_2 x(k-2) - b_1 y(k-1) - b_2 y(k-2). \tag{4.10}$$

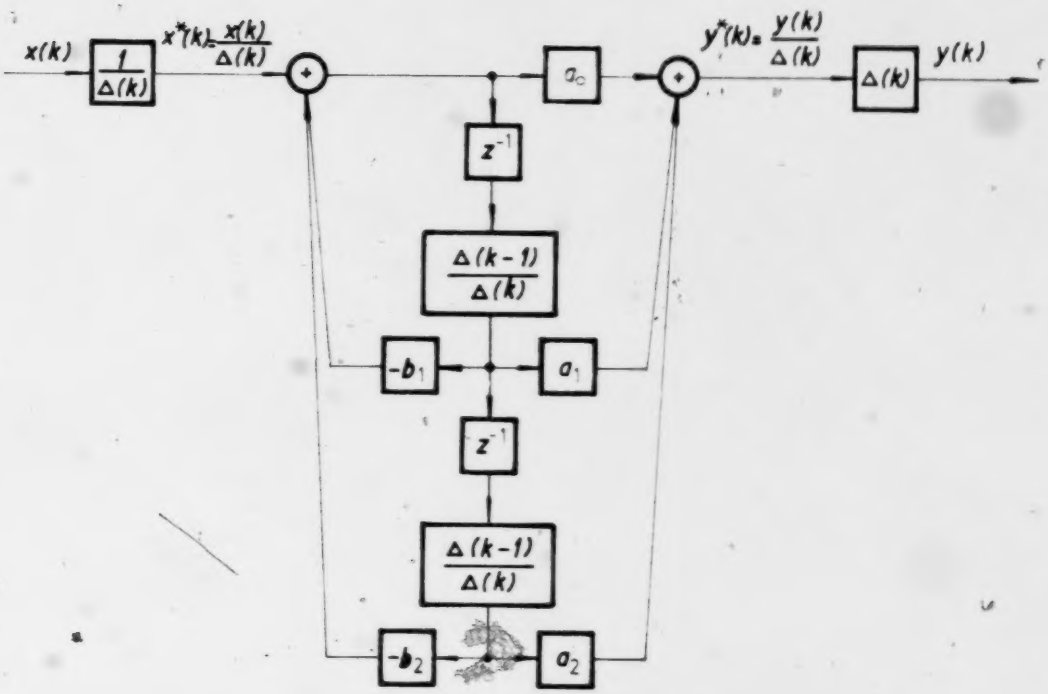
Normalizing it with respect to $\Delta(k)$ results in

$$y^*(k) = a_0 x^*(k) + [a_1 x^*(k-1) - b_1 y^*(k-1)] \cdot \frac{\Delta(k-1)}{\Delta(k)} + [a_2 x^*(k-2) - b_2 y^*(k-2)] \cdot \frac{\Delta(k-2)}{\Delta(k-1)} \cdot \frac{\Delta(k-1)}{\Delta(k)}. \tag{4.11}$$

By the same reasoning as before, a scale factor of $\Delta(k-1)/\Delta(k)$ is required at the output of each delay register so that all the computations at time k will involve in data having the same normalization factor $\Delta(k)$. The realization is as shown in Fig. 4.7 in a direct observer canonical form as well as a direct control canonical form.



(a) Direct Observer Canonical Form



(b) Direct Control Canonical Form

Fig. 4.7 Realization of a Second-Order Transfer Function with Consistent Internal Scaling.

The advantage of choosing a one-word memory step-size adaptation algorithm having step-size multipliers $\gamma=2$ and $\gamma^{-1}=1/2$ will also simplify the scaling problem of the internal data. Since the step-size at each sampling time is either increased or decreased by a factor of 2, the ratio $\Delta(k-1)/\Delta(k)$ reduces to either 2 or 1/2. This simply implies a left or right shift by one bit of the contents of all the delay registers of Figs. 4.6 and 4.7. One should note, however, that simple shifting to re-scale the register contents cannot be applied to negative numbers represented by two's complement arithmetic. Nevertheless, this can be handled quite readily by converting the negative number into its positive equivalent by two's complementing, shifting the number in the appropriate direction, and then converting back to its negative in two's complement format.

Though this quantizer can only yield a suboptimum performance, practical implementation considerations make it preferrable.

4.4.2 ADPCM Prediction Feedback Loop Scaling

The one-step-ahead plant output prediction $\bar{y}(k|k-1)$ at time k is obtained from the ADPCM predictor $P(z)$ and the control feedforward transfer function $F(z)$ based on past information available up to time $k-1$. In computing $\bar{y}(k|k-1)$, the data are normalized with respect to the previous step-size which is $\Delta(k-1)$ at time $k-1$. Thus the prediction value must be denormalized by re-scaling it by $\Delta(k-1)$ to

provide a consistently scaled output value. Also, since the prediction value is to be added to the most current quantized prediction error $\delta_q(k)$, it must be re-scaled by first denormalizing it by $\Delta(k-1)$ and then renormalized with respect to $\Delta(k)$. When $\bar{y}(k|k-1)$ is negative, it must also be converted to its positive equivalent, as discussed earlier, before it can be shifted in the re-scaling process. This approach of scaling and re-scaling will ensure that all the data in the control feedback loop are consistently normalized with respect to $\Delta(k)$ at time k. The overall scaling requirements can be summarized as shown in Fig. 4.8.

One should note that the dynamic range of the plant prediction signal $\bar{y}(k|k-1)$ is usually much wider than that of the prediction error $\delta(k)$ or the quantized prediction error $\delta_q(k)$. Therefore, the digital code-word from the A/D converter usually contributes to the least significant digits of the internal registers. In order not to lose accuracy from the A/D measurement, it is often desirable to have internal register wordlengths much longer than that of the A/D converter.

As discussed previously, the choice of the step-size multipliers to be $\gamma=2$ and $\gamma^{-1}=1/2$ can, similarly, greatly simplify the practical aspects of the implementation of the feedback control loop having an ADPCM transmitter as the quantizer. All the scaling and re-scaling problems reduce to a simple shift of the contents of the appropriate shift registers. The only problem left that one may have to pay

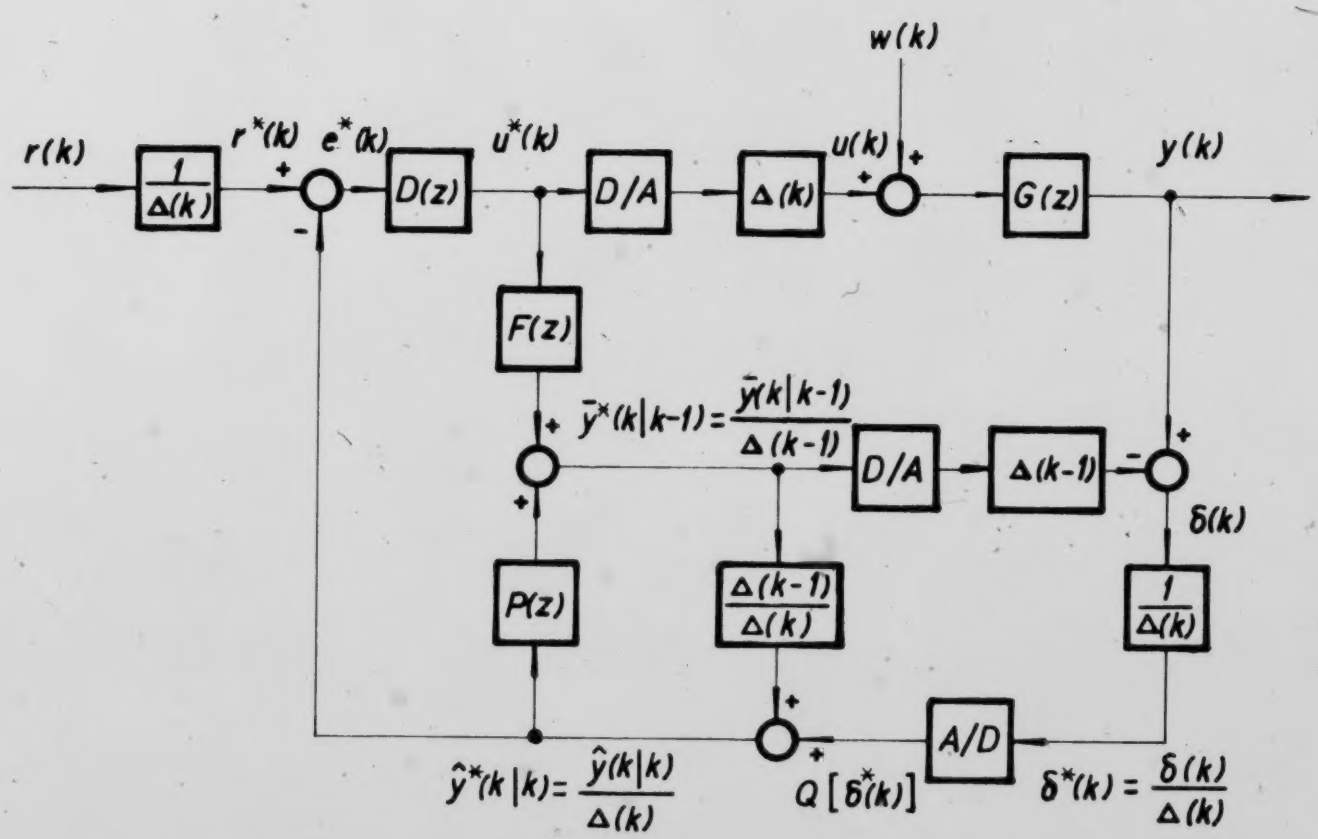


Fig. 4.8 Overall System Scaling.

particular attention to is the prevention of overflow in the intermediate arithmetic operations. This is essentially a finite wordlength effect problem and will not be studied here. The problem has been studied extensively in digital signal processing (Oppenheim and Schafer, 1975; Rabiner and Gold, 1975). It has also been studied in digital control systems (Rink and Chong, 1979; Moroney, 1983) and a more recent work is that of van Wingerden and De Konig (1984).

Chapter 5

PERFORMANCE ANALYSIS AND SIMULATION RESULTS

5.1 Introduction

The main purpose of this chapter is to evaluate the performance of the control system in Fig. 3.1 which has an ADPCM coder used as an A/D converter in the feedback path. The control system performance is evaluated from theoretical analysis as well as from digital computer simulations. However, with an adaptive quantizer in the control loop, the system becomes highly nonlinear. Theoretical nonlinear analysis will be very complicated and may not even be feasible. Therefore, the performance analysis to be presented will make use of certain assumptions so that a more tractable linear analysis may be applicable. The resulting performance estimate, though only an approximation, is still very useful for comparing with the performance results from computer simulations. In addition to the approximate performance analysis, a brief discussion on the stability of the closed-loop control system having an ADPCM A/D converter will also be included. Finally, computer simulation results for two specific examples are presented to illustrate the benefits of an ADPCM A/D in a control system and to verify some of the design criteria derived in previous chapters.

5.2 Performance Analysis

The performance of the control system shown in Fig. 3.1 depends on the design of the LQG compensator, the process disturbance, and the errors due to the ADPCM quantization noise introduced in the feedback path. In order to evaluate the tracking performance of the control system, transfer functions from r, w, ϵ to the plant output y are required. The presence of the adaptive quantizer in the control loop makes the system highly nonlinear. In order to obtain the transfer functions, certain assumptions are necessary. In the following performance analysis, it will be assumed that the adaptive quantizer is well designed and properly scaled such that quantizer saturation and slope overload rarely occur. Therefore, granular noise will be the dominant source of quantization error and hence it can be modelled as an additive noise. Furthermore, it will be assumed that steady-state has been reached and the quantizer will employ a constant steady-state step-size Δ_{ss} for a given process disturbance power level R_w. As a result, the quantization error can be assumed to be a zero-mean, stationary, and uniformly distributed random sequence. This approach of analysis is, however, only an approximate and simplified way of evaluating the control system performance. It is essentially an analysis for a nonadaptive DPCM system which has been scaled properly for a given value of R_w. Nevertheless, it reduces the nonlinear analysis problem to a much simpler, linear one, and the resulting performance

estimate can provide a useful measure for evaluating the performance benefits of an ADPCM coder used as an A/D converter in a control system.

Consider the block diagram in Fig. 3.1. Assuming a deterministic problem for the moment, the plant output is given by

$$y(z) = G(z)w(z) + G(z)u_o r(z) + D(z)G(z)[r(z) - \hat{y}(z)], \tag{5.1}$$

where it has been assumed that a steady-state offset compensation control term, $u_o r$, defined in (B.1), is also included in the control signal u. From (B.12), the feedback signal $\hat{y}$ is given by

$$\hat{y}(z) = \frac{D(z)G(z)}{1 + D(z)G(z)} r(z) + \frac{G(z)}{1 + D(z)G(z)} u_o r(z) + \frac{\delta_q(z)}{[1 - P(z)][1 + D(z)G(z)]}. \tag{5.2}$$

Substituting (5.2) into (5.1), one would get

$$y(z) = \frac{D(z)G(z)}{1 + D(z)G(z)} r(z) + \frac{G(z)}{1 + D(z)G(z)} u_o r(z) + G(z)w(z) - \frac{D(z)G(z)}{[1 - P(z)][1 + D(z)G(z)]} \delta_q(z). \tag{5.3}$$

The output tracking error is then given by

$$e_o(z) = r(z) - y(z)$$

$$= \frac{1 - G(z)u_o}{1 + D(z)G(z)} r(z) - G(z)w(z)$$

$$+ \frac{D(z)G(z)}{[1 - P(z)][1 + D(z)G(z)]} \delta_q(z). \qquad (5.4)$$

The first term in (5.4) describes the closed-loop dynamic behaviour of the plant output error due to the reference input or set-point changes. For a Type-0 system with the u_o offset compensation control term or a Type-1 system, this term will vanish when the plant output has reached the set-point after the transients due to the set-point changes have settled down. Neglecting this term, the output tracking error caused by the quantized prediction error and the process disturbance is given by

$$e_o(z) = \frac{D(z)G(z)}{[1 - P(z)][1 + D(z)G(z)]} \delta_q(z) - G(z)w(z). \qquad (5.5)$$

With the adaptive quantizer "homed into" a constant step-size Δ_{ss} at steady-state for a given R_w value, together with the additive quantization noise assumption, then $\delta_q(z)=\delta(z)+\epsilon(z)$ and $\delta(z)$ is given by (B.15). Substituting these into (5.5) and making use of (3.23) yields

$$e_o(z) = \frac{D(z)G(z)}{1 + D(z)G(z)} \epsilon(z) - \frac{G(z)}{1 - D(z)G(z)} w(z). \qquad (5.6)$$

This is simply the steady-state closed-loop response to the

additive feedback measurement noise ϵ and the process disturbance w. The same equation also applies to a conventional digital feedback control system which does not have the ADPCM feature but only a fixed A/D converter in the feedback path. Relaxing the deterministic assumption and assuming, as before, that ϵ and w are mutually uncorrelated zero mean, white random sequences, the steady-state output tracking mean-square-error (MSE) is then given by

$$e_o(z) = \frac{1}{2\pi j}\oint_{|z|=1}\left|\frac{D(z)G(z)}{1 + D(z)G(z)}\right|^2\frac{dz}{z}\cdot E[\epsilon^2] + \frac{1}{2\pi j}\oint_{|z|=1}\left|\frac{G(z)}{1 + D(z)G(z)}\right|^2\frac{dz}{z}\cdot E[w^2]$$

$$= G_c \cdot R_\epsilon + G_w \cdot R_w, \qquad (5.7)$$

where it has been defined that

$$G_c = \frac{1}{2\pi j}\oint_{|z|=1}\left|\frac{D(z)G(z)}{1 + D(z)G(z)}\right|^2\frac{dz}{z};\ R_\epsilon = E[\epsilon^2], \qquad (5.8)$$

and

$$G_w = \frac{1}{2\pi j}\oint_{|z|=1}\left|\frac{G(z)}{1 + D(z)G(z)}\right|^2\frac{dz}{z};\ R_w = E[w^2]. \qquad (5.9)$$

Normalizing (5.7) with respect to R_w, one can then obtain a normalized steady-state output tracking MSE

$$\frac{E[e_0^2]}{R_w} = G_c \frac{R_\epsilon}{R_w} + G_w$$

$$= G_c \frac{1}{R} + G_w, \qquad (5.10)$$

where R is defined as before in (3.46) and is simply R_{ss}, the solution to (3.52), which has already been specified when designing the Kalman filter predictor $P(z)$ and $F(z)$ as well as the quantizer scaling as discussed before. Since G_c and G_w involve the Kalman filter, which have already been determined by specifying R_{ss} and the LQG compensator, (5.10) can be evaluated quite readily by numerical methods once the LQG weighting factor ρ and hence the optimal state feedback gain K is specified.

As can be seen from (5.10), this performance evaluation can be specified completely during the design stage when the Kalman filter and the LQG compensator are specified. It serves only as a lower performance limit which can be expected from a nonadaptive DPCM coder that has been scaled properly for each given R_w value in a feedback control application. The actual performance will be different from the estimate obtained from (5.10) because occurrence of quantizer saturation and slope overload errors will increase the MSE. Furthermore, the adaptive nature of the quantizer and its signal-dependent as well as sequentially correlated quantization error is not accounted for in the analysis. Nevertheless, the result of (5.10) is still useful for evaluating the performance of a control system having an

ADPCM A/D converter for a given range of R_w values in which the adaptive quantizer can adapt its step-sizes. For example, one may regard the performance estimate of (5.10) as an "optimum" performance index. Then, the performance of a control system with ADPCM feature can be compared with this index to determine how well the control system performance can match up to this optimum value.

When quantizer saturation and slope overload occur, however, the preceding additive granular noise assumption cannot be applied. The quantized prediction error δ_q can no longer be separated into δ and ϵ. In this situation, the steady-state output tracking error e_o is given by (5.5) in which the term $1/[1-P(z)]$ can be shown to contain open-loop plant poles. Hence, (5.5) is essentially the uncontrolled open-loop response to the process disturbance w and the saturated quantizer output δ_q. The prediction error δ in this situation will be given by (B.13) which also contains open-loop plant poles (see Eq.(3.20)) and is essentially an open-loop response to the quantization noise ϵ and the plant disturbance w. Consequently, if the plant contains unstable open-loop poles, instability of the control system and the prediction error may also occur. But, with an adaptive quantizer which can adapt its step-sizes, it is possible to avert the occurrence of instability. When quantizer saturation occurs, the quantizer can increase its step-size by the factor γ so that the system can catch up at the next sampling period before slope overload develops. Then the

quantizer input will be within the quantizer range again so that granular noise will once again be the dominant source of quantization error. When slope overload occurs, the quantizer will open up by increasing its step-sizes by the factor γ at each sampling period until it catches up and recommences tracking of the quantizer input signal. In this case, however, the ability of the quantizer to catch up depends on the rate at which the quantizer input, which is the unstable δ signal, is increasing as well as how fast the quantizer can open up during slope overload.

Thus it can be seen that proper selection of the adaptive quantizer parameters is very crucial to the performance of the control system. Inadequate design may even lead to detrimental effects such as instability and loss of control of the system. The simulation results presented in the following sections will illustrate these issues.

5.3 Computer Simulation Results

The performance of an ADPCM coder in a control loop was evaluated by digital computer simulations of the system shown in Fig. 3.1. Two particular second-order examples were selected to be the controlled plants in these simulations. One of them was a marginally stable process and the other was an unstable plant. These systems were chosen beacuse in addition to revealing the merits of the ADPCM coder when it was used as an A/D converter, they were also useful for

demonstrating the stability aspect of such a coder in a control system application. The transfer functions $G(z)$, $D(z)$, $P(z)$, and $F(z)$ were all simulated on a general purpose digital computer having a very long wordlength and so internal finite wordlength effects are negligible. The random sequence w was generated from a zero mean, unit variance Gaussian random number generator and was scaled by R_w to obtain a process disturbance which has the desired variance value. The D/A converters in Fig. 3.1 were assumed to have a very long wordlength which was compatible to the computer internal register wordlength. They were also assumed to have been properly scaled. Hence, no further quantization error was introduced by these converters. The only source of quantization error is essentially the errors introduced by the adaptive quantizer of the ADPCM coder in the feedback loop.

The adaptive quantizer used in the ADPCM coder in the simulations has a structure similar to that shown in Figs. 4.1 and 4.2. Basically, it is made up of a fixed quantizer which is pre-scaled by a gain factor $1/D(k)$ where $D(k)$ is a scaling factor directly proportional to the step-size $\Delta(k)$. The output of the quantizer was then re-scaled by the factor $D(k)$. The resulting quantized signal is equivalent to one which has been adaptively quantized. In the simulations the step-size adaptation parameter was chosen to be $\gamma=2$, which is a very practical value for hardware implementations of the system shown in Fig. 3.1.

With $\gamma=2$ in the adaptation algorithms, the scaling factor $D(k)$ used in the simulations was set up in such a way that

$$D(k) = \begin{cases} 1, & \text{when } \Delta(k)=\Delta_{max} \\ 2D(k-1), & \text{when } \Delta(k)=2\Delta(k-1) \\ D(k-1)/2, & \text{when } \Delta(k)=\Delta(k-1)/2 \\ 2^{-2M}. & \text{when } \Delta(k)=\Delta_{min} \end{cases} \tag{5.11}$$

Therefore, whenever $\Delta(k)<\Delta_{max}$, the quantizer input would be pre-scaled by the gain factor $1/D(k)$ from the allowable set $\{2,2^2,2^3,\ldots,2^{2M}\}$, and the output of the fixed quantizer would be re-scaled by the factor $D(k)$ from the allowable set $\{2^{-1},2^{-2},2^{-3},\ldots,2^{-2M}\}$. When $\Delta(k)=\Delta_{max}$, then $D(k)=1$ and the quantizer input would simply be applied directly to the fixed quantizer. This implies that the fixed quantizer must be scaled in such a way that its step-size would be the same as Δ_{max}, which is obtainable from (3.54) for a given $R_{W_{max}}$ value, or from $\Delta_{max}=2^M\Delta_0$ where Δ_0 is given by (3.55) for a specifed R_{W_0} value. Both (3.54) and (3.55) require R_s from the solution of (3.52) for a given number of quantizer bits b and a given quantizer loading factor. In the simulations, R_s was solved from (3.52) by numerical and graphical means, as illustrated later, for quantizers which were 3σ and 4σ loaded and with $b=3$ as well as $b=4$. Δ_{max} was specified by (3.54) for an assumed worst case maximum value of $R_{W_{max}}$.

To evaluate the performance of the control system in the simulation, two different performance indices were used. The first was the normalized steady-state output tracking

mean-square-error (MSE) similar to that in (5.10) for a wide range of R_w variance values. It was calculated by averaging the squares of 5000 consecutive samples of the steady-state output tracking error and normalizing it with respect to the process disturbance variance R_w:

$$\frac{E[e_0^2]}{R_w} = \frac{1}{R_w} \cdot \frac{1}{5000} \sum_{k=100}^{5100} [r(k) - y(k)]^2. \qquad (5.12)$$

The reference signal r in these simulations was a constant unity set-point (S.P.) applied to the system at time $k=0$. That is, it is simply a unit step input. Thus the performance index (5.12) is simply the performance of the unit step response of the control system at steady-state when all the transients due to set-point changes had died down. This performance can then be compared with the performance estimate from (5.10) to evaluate how well the ADPCM control system can match up to this "optimum" performance index.

The other performance index used in the performance evaluation was similar to (5.12) except that it was not normalized with respect to R_w and the reference input was not a constant set-point. Instead, for every 100 sampling periods, a random set-point values was generated by a random number generator which is uniformly distributed over the interval $(0, r_{max})$. The output tracking mean-square-error was calculated from

$$E[e_0^2] = \frac{1}{5000} \sum_{k=100}^{5100} [r(k) - y(k)]^2 \qquad (5.13)$$

for a wide range of R_W values. This performance indicator is useful for evaluating the transient performance of the control system for a random set-point which is randomly changing periodically for every 100 samples.

Besides the output tracking error performance indices, another indicator was also used in the system performance evaluation. It was the root-mean-square (RMS) value of the control signal, U_{rms}, and was computed by averaging the squares of 5000 consecutive samples of the control signal and then taking the square root of the resulting average value:

$$U_{rms} = \sqrt{\frac{1}{5000} \sum_{k=100}^{5100} U^2(k)} \; . \qquad (5.14)$$

Together with (5.12) and (5.13), (5.14) is a very useful index for evaluating the benefits of applying an APDCM coder as an A/D converter in a control system.

In the following sections, simulation results are presented graphically for various combinations of the ADPCM quantizer parameters. The results of computing (5.12), (5.13) and (5.14) from the simulation results are shown as points on the graphs and are also connected by straight line segments. The steady-state performance estimates of (5.10) are shown on the appropriate graphs as lines which are indicated by the symbol ◄.

5.3.1 ADPCM in a Marginally Stable Plant

The marginally stable second-order plant (Plant A) used in the simulations has a transfer function (including zero-order-hold) given by,

$$G(z) = \frac{Y(z)}{U(z)} = \frac{a_1 z + a_2}{z^2 - b_1 z - b_2}, \qquad (5.15)$$

where a_1=0.04837418, a_2=0.0467884, b_1=1.904837418, and b_2=-0.904837418. This will result in open-loop plant pole locations at z_1=0.9048374 and z_2=1.000. Because of the presence of the integrator pole at z_2=1.000, this plant is a Type-1 system and hence the closed-loop system will have zero steady-state error to step reference inputs (constant set-points).

The transfer function (5.15) can be realized in a direct observer canonical form similar to that shown in Fig. 4.7a and then formulated into a state-space form by assigning the outputs of the delay elements as the states of the system. The resulting state-space equation is given by

$$\mathbf{x}(k+1) = \begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} b_1 & 1 \\ b_2 & 0 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} u(k) + \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} w(k)$$
$$= \Phi \mathbf{x}(k) + \Gamma u(k) + \Gamma_1 w(k), \qquad (5.16)$$

where it has been assumed that the process disturbance appears at the plant input and hence $\Gamma_1=\Gamma$. The state transition matrices of the system are given by

$$\Phi = \begin{bmatrix} b_1 & 1 \\ b_2 & 0 \end{bmatrix} \text{ and } \Gamma_1 = \Gamma = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} . \qquad (5.17)$$

The plant output is given by

$$y(k) = x_1(k) = \mathbf{H}\mathbf{x}(k), \qquad (5.18)$$

where $\mathbf{H}=[1\ 0]$. This state-space equation can then be applied to the steady-state Kalman filter equation to solve for the steady-state gain vector $\mathbf{L}$.

To solve for the steady-state variance ratio R_s, the left-hand side (LHS) and the right-hand side (RHS) of Eq.(3.52) must be evaluated. The LHS can be calculated quite readily when the quantizer loading factor and the number of quantizer bits b are specified. For 3σ loading, $a=1/3$, and for 4σ loading $a=3/16$ in (3.52). The RHS depends on the steady-state Kalman filter gain $\mathbf{L}$ as well as the variance ratio R. The steady-state gain $\mathbf{L}$ can be evaluated numerically by computing the eigenvalues and eigenvectors, as a function of R, of the Hamiltonian equation for this system (Franklin and Powell, 1980). By specifying an R value, $\mathbf{L}$ can be calculated and then substituted into $F(z)$ and $P(z)$. The gains G_r and G_p in (3.52) require evaluations of the complex integrals in (3.43) and (3.44), respectively. These can be computed quite readily by a numerical procedure described by Åström *et al.* (1970). The numerical values of the LHS and the RHS of (3.52) can then be plotted against various values of R as shown in Fig. 5.1. The intersections

of the curves on the plot will yield the solution R_s, which can then be used to compute the required steady-state Kalman filter gain L. Furthermore, R_s is also used to calculate the step-size of the quantizer. In the simulations, with a given loading factor and number of quantizer bits, Δ_{max} was calculated from (3.54) with R_s obtained from Fig. 5.1 for the given conditions and an assumed maximum value of $R_{W_{max}} = 10^3$.

The steady-state optimal state feedback gain vector K can be computed as a function of the LQG weighting factor ρ by solving for the eigenvalues and eigenvectors of the

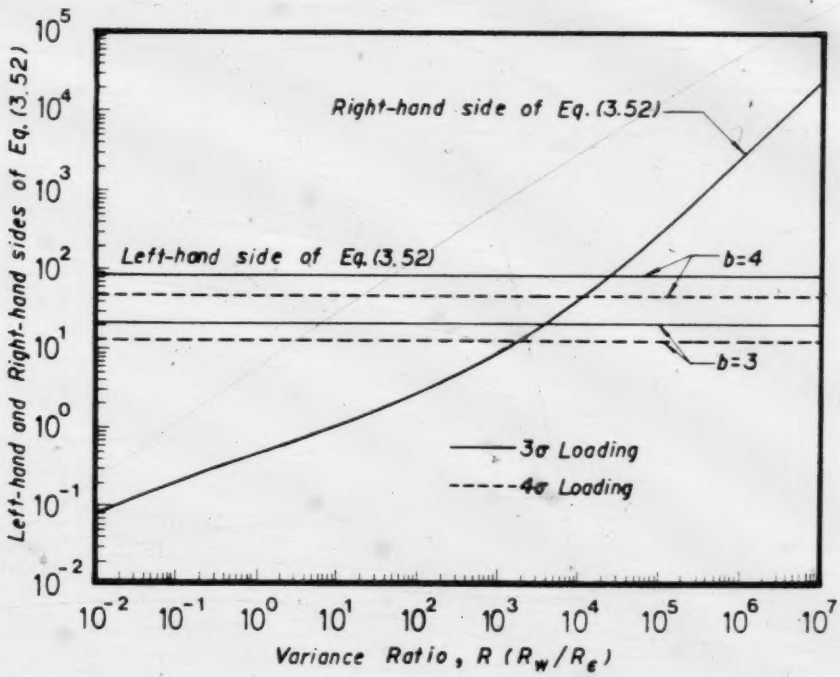


Fig. 5.1 Graphical Solution of Eq.(3.52) for Plant A.

Hamiltonian equations, similar to the Kalman filter problem, for the optimal control problem (Franklin and Powell, 1980). Once **K** is calculated, together with the gain **L** calculated when R_e is specified, the LQG compensator transfer function (3.10) will then be specified completely. With this specified LQG compensator transfer function, step responses were simulated for various ρ values with R_e specified from Fig. 5.1 for a 3σ loaded and b=4 quantizer situation. Fig. 5.2 illustrates the resulting unit step responses, at the sampling instants, of Plant A in an ideal unity feedback control system where there are no quantization error and no

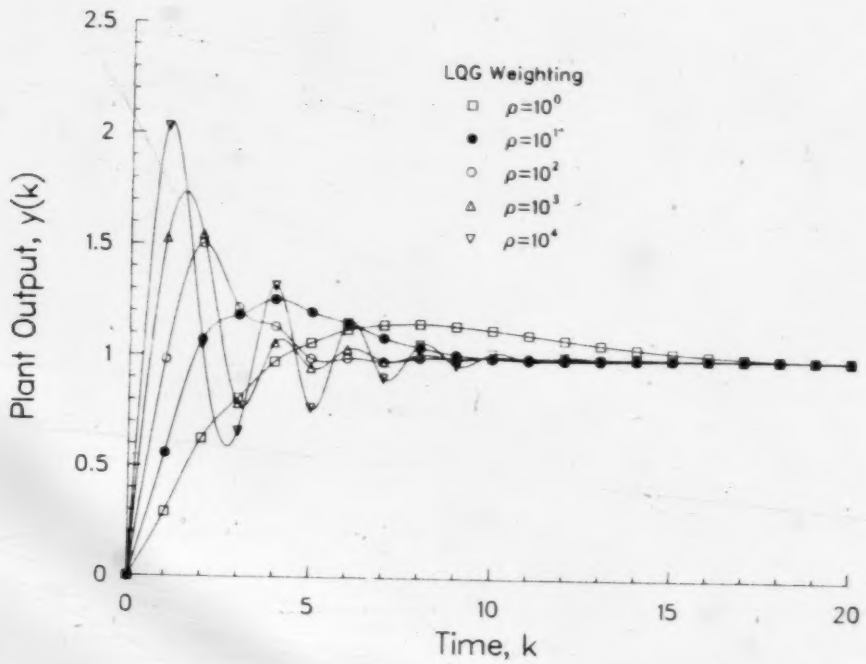


Fig. 5.2 Closed-Loop Unit Step Responses of Plant A for Various LQG Weighting Factors ρ.

process disturbance. It can be seen from Fig. 5.2 that for low ρ weightings, the step response is highly damped and takes a relatively long time to reach the steady-state. For higher ρ values, however, the step response is very oscillatory during the transient and takes a relatively short time to reach the steady-state. The selection of a suitable ρ is determined by the allowable overshoot during the transients and the time required to reach the steady-state. It is also determined by the loss function (3.9) and the allowable control efforts. The loss function (3.9) will imply a large control signal for high ρ values and vice versa.

The steady-state performance estimate (5.10) can also be evaluated when the LQG compensator $D(z)$ is specified. The gains G_c and G_w in (5.10) are readily obtainable by evaluating the complex integrals in (5.8) and (5.9), respectively, by the numerical procedure mentioned earlier. Fig. 5.3 illustrates the results of computing (5.10) for various values of ρ with R_s specified from Fig. 5.1 for a 3σ loaded and b=4 quantizer as in Fig. 5.2. It is apparent from Fig. 5.3 that with a higher ρ weighting, one would expect a lower steady-state output tracking error from the LQG control system.

In the following sections, steady-state output tracking error performances for various ADPCM parameters and configurations are presented in graphical form as explained earlier.

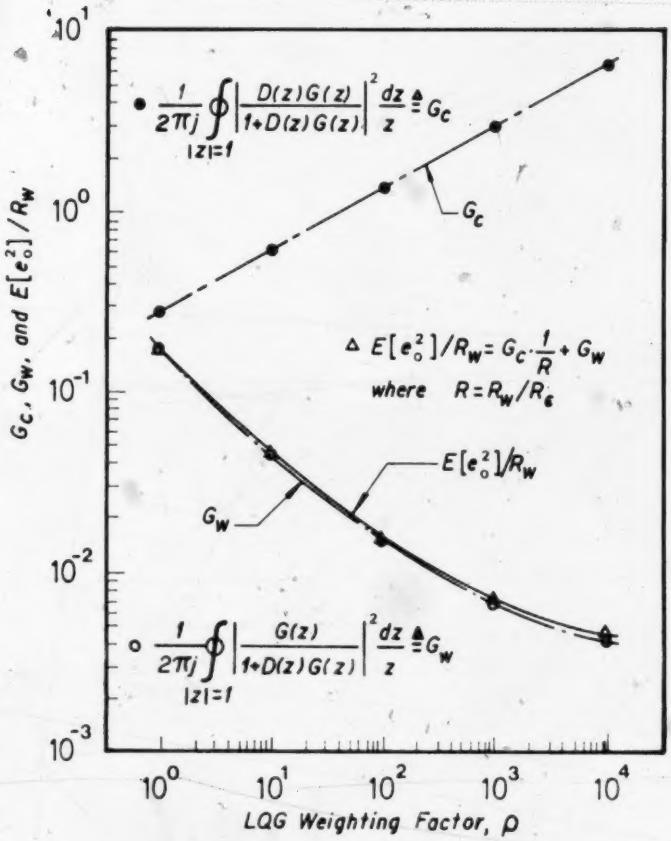


Fig. 5.3 Performance Estimate Eq.(5.10) for Plant A with b=4 and 3σ Loading.

5.3.1.1 Performance Versus c

Figs. 5.4 to 5.8 illustrate the steady-state control system performance with unit step input (unity set-point (S.P.)) for various c values, selected from the allowable set (3.32), in the adaptation algorithm (3.30). The A/D converter in the simulation was a feedback ADPCM (APDCMQ) coder which was 3σ loaded with b=3 and b=4. M in (3.30) was chosen to be M=4 to give a wide quantizer adaptation range (48 dB) which is given by $R_d=2^{2M}$. The LQG weghting factor ρ

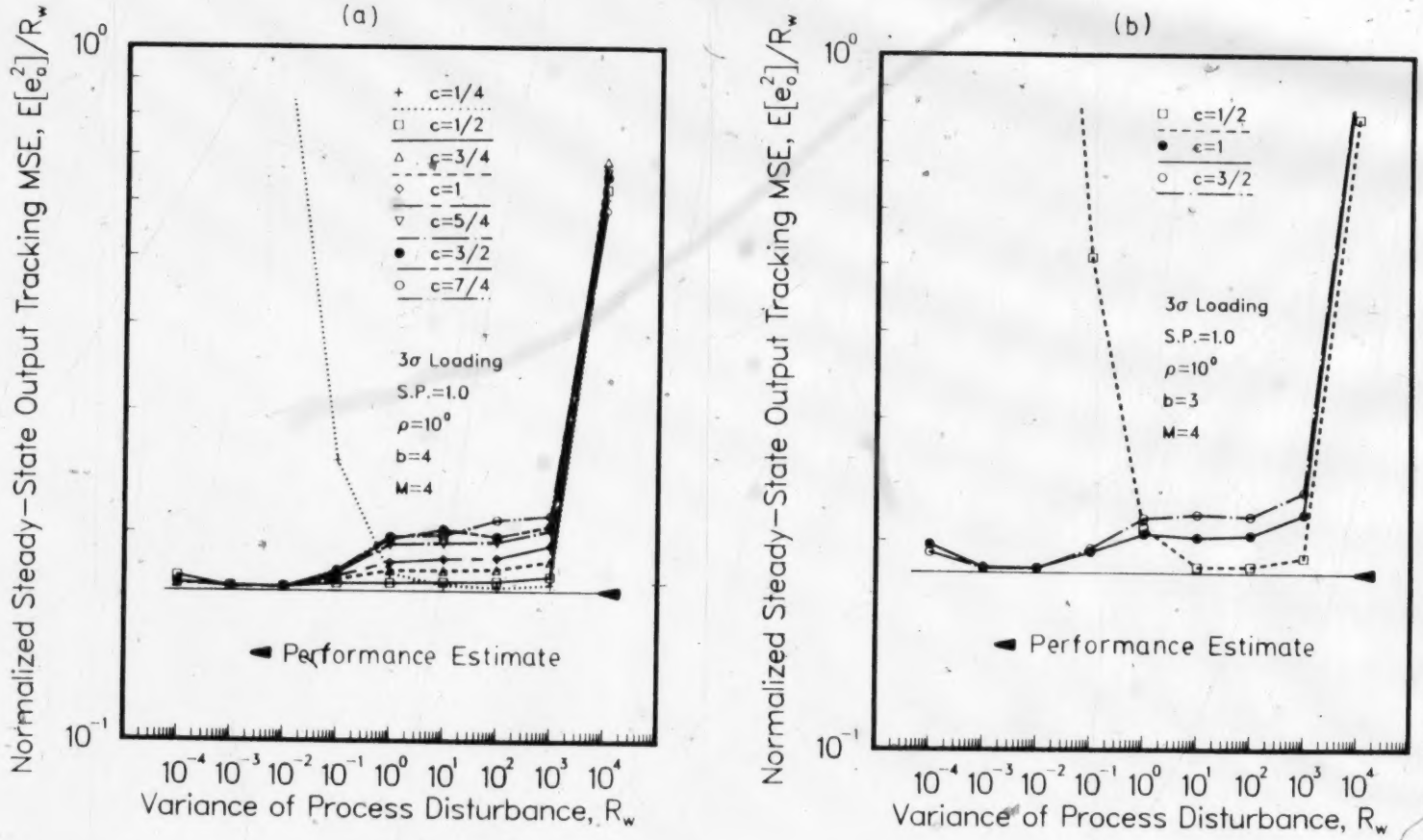


Fig. 5.4 Steady-State Mean-Square-Error for $\rho=1$ and Various c values. (a)b=4 Bits. (b)b=3 Bits.

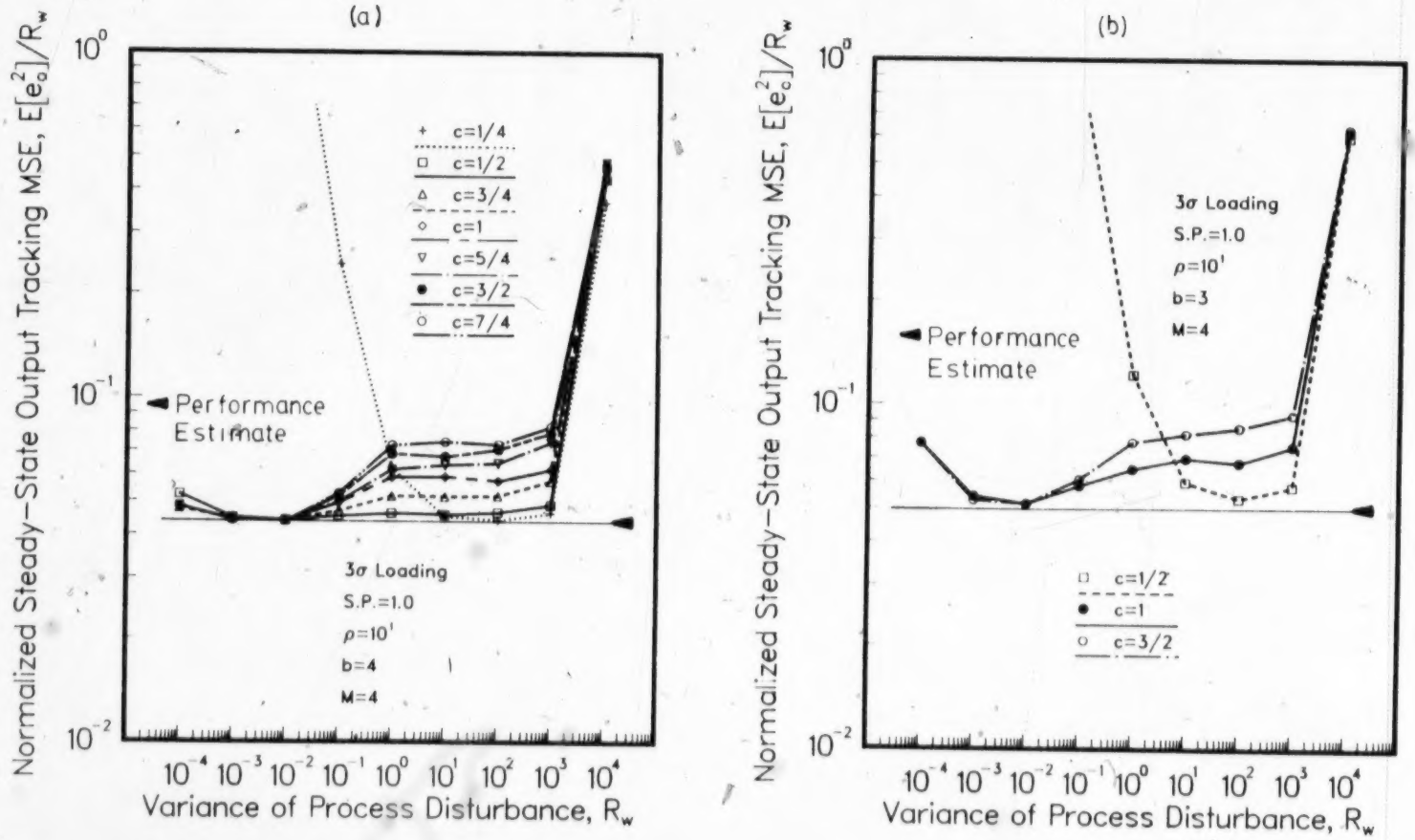


Fig. 5.5 Steady-State Mean-Square-Error for $\rho=10$ and Various c values. (a)b=4 Bits. (b)b=3 Bits.

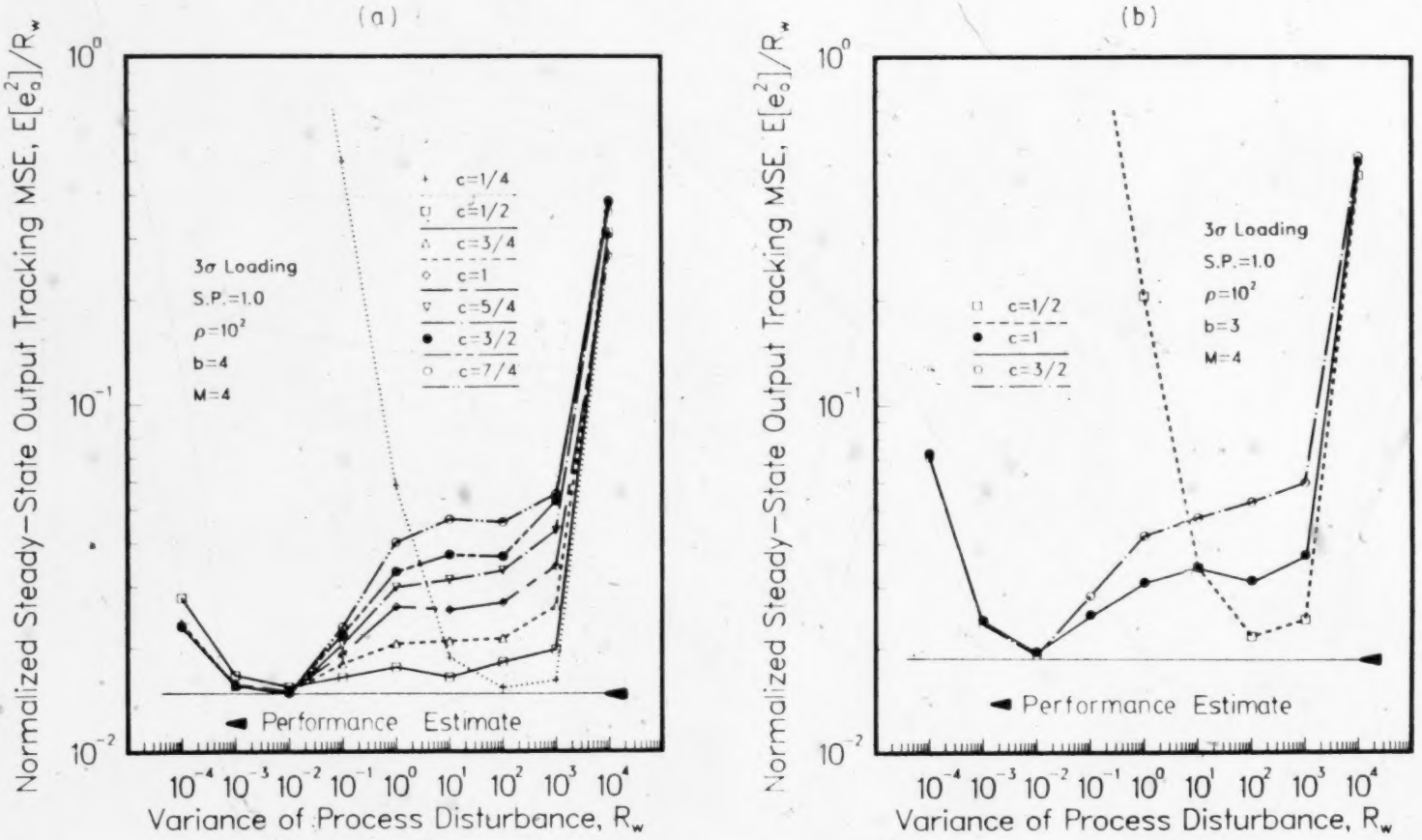


Fig. 5.6 Steady-State Mean-Square-Error for $\rho=10^2$ and Various c values. (a)b=4 Bits. (b)b=3 Bits.

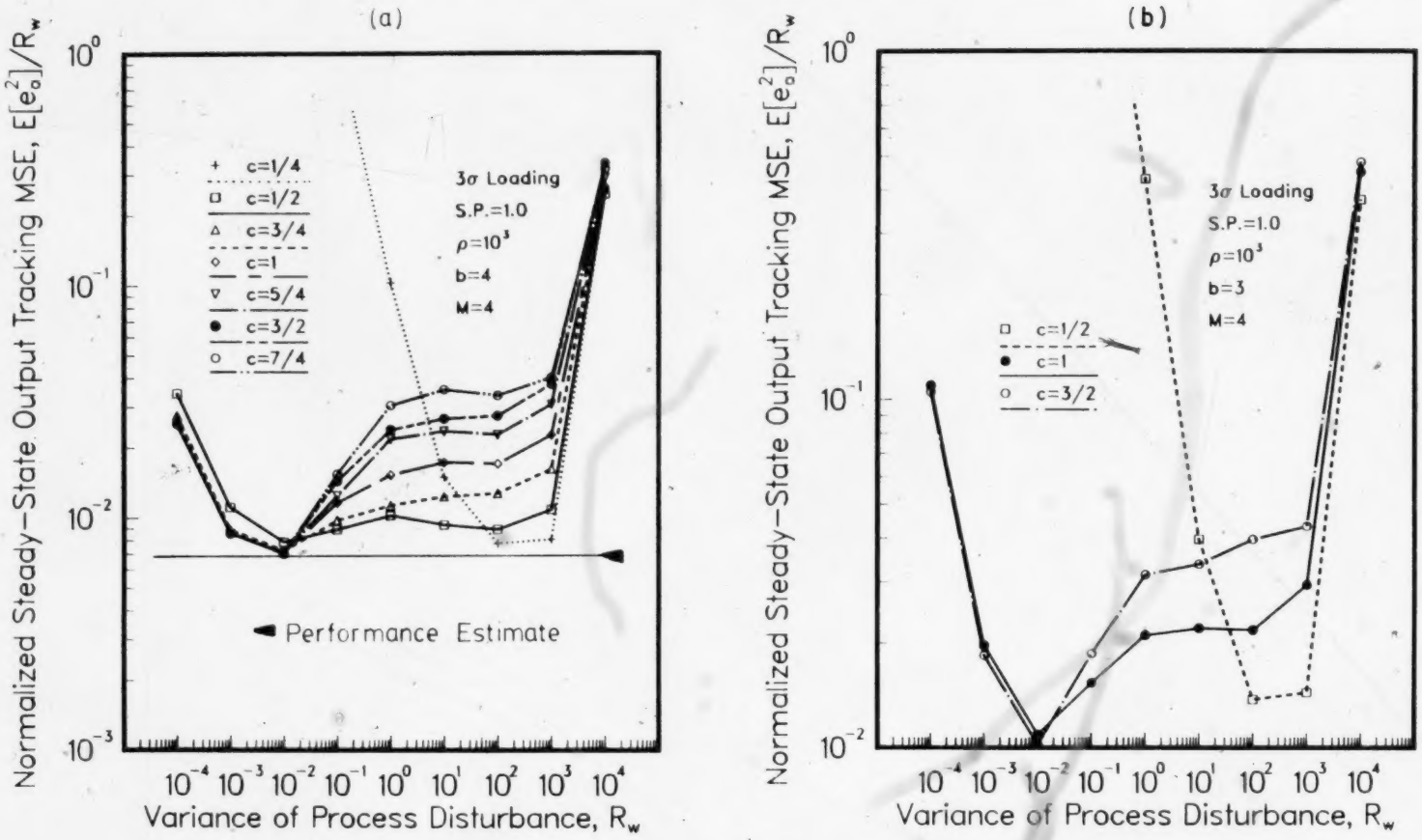


Fig. 5.7 Steady-State Mean-Square-Error for $\rho=10^3$ and Various c values. (a)b=4 Bits. (b)b=3 Bits.

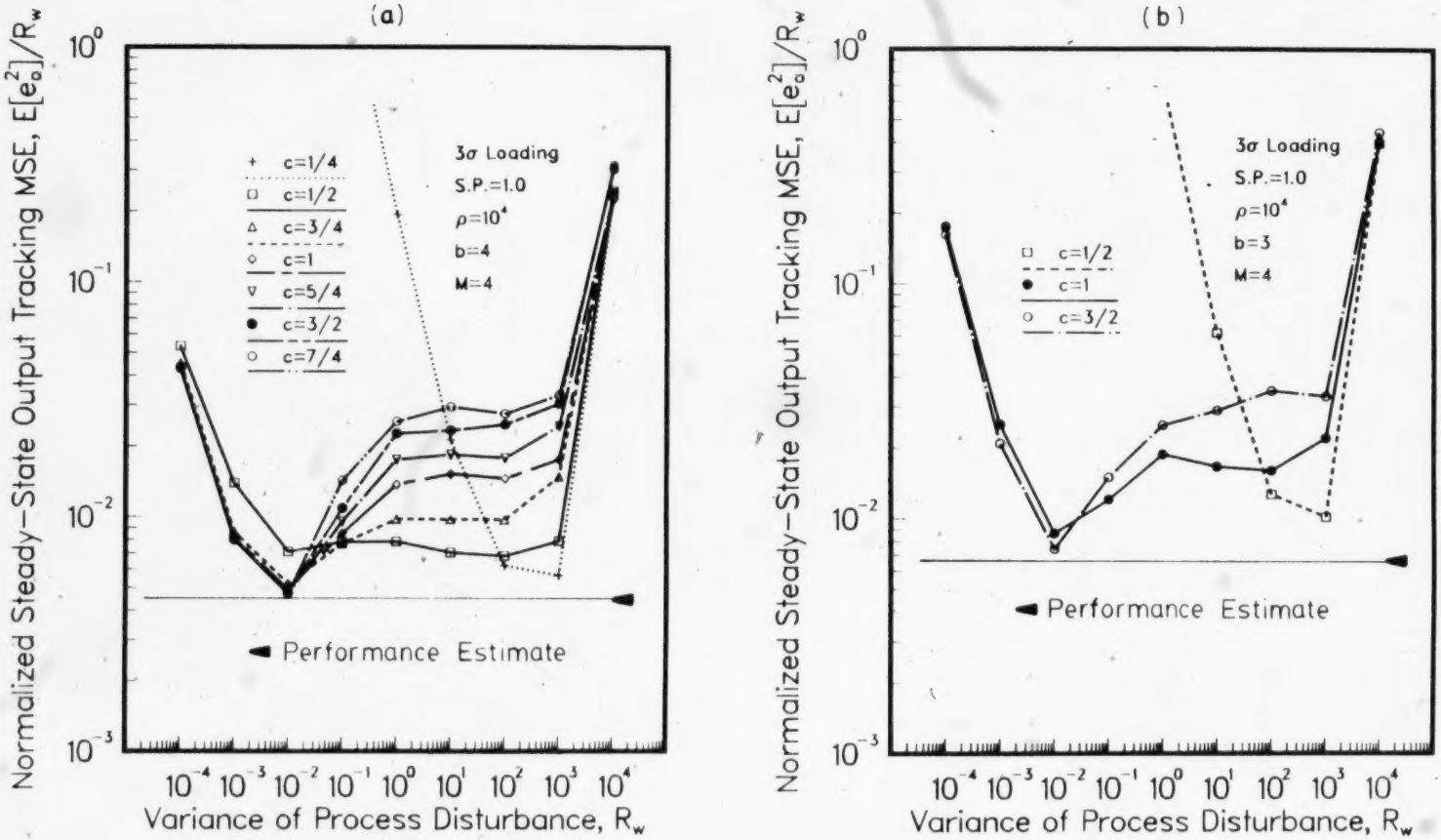


Fig. 5.8 Steady-State Mean-Square-Error for $\rho=10^4$ and Various c values. (a)b=4 Bits. (b)b=3 Bits.

ranged from 1 to 10^4 so that the unit step responses of the closed-loop system would range from very slow to very fast (see Fig. 5.2). In the simulation, R_w ranged from 10^{-4} to 10^4 in 10 dB steps. This wide range is very useful for examining the ability of the quantizer to adapt to different environments. Note that when $R_w=10^4$, the system would be driven by a process disturbance having a variance value beyond the maximum that can be handled by the adaptive quantizer.† This is a useful test for stability when quantizer saturation and slope overload occur frequently in the ADPCM coder, resulting in frequent loss of feedback information. As shown before in (5.5), under such conditions, closed-loop instability may result if the plant contains unstable open-loop poles. The performance estimate (5.10) for the steady-state tracking error was also computed and is shown on Figs. 5.4 to 5.8 as a straightline indicated by ◄.

In general, as evident from Figs. 5.4 to 5.8, the control system having an ADPCM A/D converter performs very well and is robust to a wide range of R_w values. The control system performance is, however, greatly affected by the selection of c in the adaptation algorithm (3.30). With $\gamma=2$, Eq.(3.39) implies that $c\geq 1/2$ for $b=4$ and $c\geq 1$ for $b=3$. This is verified by the simulation results. As apparent from Figs. 5.4 to 5.8, when c is below the limiting values, the performance greatly deteriorates for low R_w values. This is

---

†Δ_{max} was specified from (3.54) for a maximum value of $R_{w_{max}}=10^3$.

a result of an increase in quantization noise when c is too low. As discussed before, this causes C_r to be too low and hence a step-size larger than the optimum value will be used. From the figures, when c is below the limit set by (3.39), the robustness of the system to R_w will be lost and hence the adaptive quantization feature of the ADPCM coder will not be very effective for low R_w levels.

Comparing the figures, it is seen that when $c>1/2$ ($b=4$) and $c>1$ ($b=3$), the performance deteriorates at higher R_w values. This illustrates the fact that when c is too large, C_r will be too high and this will imply a step-size smaller than the optimum value. As a result, slope overload and quantizer saturation occur more frequently. Thus the quantization noise is increased significantly and hence the performance is greatly affected. The simulation results in the figures suggest that c be selected to be the limiting value of (3.39) to give the best compromise between robustness to R_w and consistency of steady-state tracking error performance. These results have also shown that the selection of the c parameter is not very sensitive to the different LQG weighting factors. For all ρ factors, the best c is the limit of (3.39).

For $b=4$ and with $c=1/2$, except for $\rho=10^3$ and 10^4, the control system performance agrees reasonably well with the performance estimate of (5.10). In all cases, the lowest MSE value occurs when $R_w=10^{-3}$. This value is the lower limit,

$R_{W_{min}}$, that can be handled by the adaptive quantizer.† With this R_W value, the quantizer is essentially a fixed quantizer since Δ_{min} would be used most of the time. When $R_W<10^{-2}$, Δ_{min} is too large and hence the resulting quantization noise will be increased and subsequently the control system performance will deteriorate. On the other hand, when $R_W>10^{-2}$, the adaptive quantizer is effective and hence will inject a signal-dependent quantization noise into the system. Then, the additive quantization noise assumption is no longer valid and thus the linear transfer function approach of analysis becomes grossly inadequate. Furthermore, the LQG compensator is designed with white system noise assumptions. Hence the controller is not capable of handling the sequentially correlated quantization noise, which arises because of the step-size adaptation. With the presence of the correlated quantization noise in the LQG control system, the performance will generally deterioriate (Moroney, 1983). Therefore, a discrepancy exists between the performance of the ADPCM coder and the theoretical estimate (5.10) for a nonadaptive DPCM which has been properly scaled for each R_W value. One can also notice that when R_W is near the maximum $R_{W_{max}}$ value, the simulation results deviate further away from the theoretical estimated values. This is the result of the more frequent occurrences

---

†With M=4, which will yield an adaptation range of about 48 dB, the adaptive quantizer can handle R_W values ranging from 10^{3} down to 10^{-2} when Δ_{max} is designed for the assumed $R_{W_{max}}$ value.

of quantizer saturation and slope overload when R_W is at higher values.

For b=3 and with c=1, the best MSE also occurs when $R_W=10^{-2}$ at which the simulation results agree well with the performance estimate (5.10). As mentioned earlier, the adaptive quantizer at this R_W is essentially a fixed quantizer having a step-size of Δ_{min}. For $R_W<10^{-2}$, Δ_{min} is too large and thus the tracking error becomes relatively large. For $R_W>10^{-2}$, the control system performance begins to deviate from the performance estimate. This deviation becomes larger when R_W is approaching $R_{W_{max}}$ as well when ρ is increasing. This deviation is also observed to be larger than that of b=4. This shows the disadvantage of the coarse quantization resulting from using only 3 bits of quantizer wordlength that can provide only 4 quantizer output levels (see Fig. 3.2). As a result, the adaptive quantizer is driven into quantizer saturation and slope overload more often when compared with the b=4 quantizer, which has more output levels. Morever, comparing (a) and (b) of Figs. 5.4 to 5.8, it can be seen that the system with b=4 always outperforms the one with b=3 bits. Thus it is perhaps preferrable to use a b=4 bit quantizer to give finer quantization than a b=3 bit quantizer which will result in very coarse quantization. In general, one can expect that with more quantizer bits in the ADPCM coder, better control system performance can be obtained.

When $R_W=10^4$, which is beyond the assumed maximum value of $R_{W_{max}}=10^3$, the steady-state output tracking MSE is still bounded for both the b=3 and b=4 quantizers. With $R_W>R_{W_{max}}$, the dynamic range of the quantizer is not adequate. The quantizer under such condition will be driven into saturation and slope overload, resulting in loss of feedback information. Then, the control system is essentially in an open-loop uncontrolled condition. The performance depends on the ability of the open-loop plant to suppress the process disturbance and on the stability of the open-loop plant poles. For this marginally stable Plant A, no instability was observed in the simulations when $R_W=10^4>R_{W_{max}}$. The steady-state error is still bounded as shown in Figs. 5.4 to 5.8, although it deviates from the performance estimate significantly.

In the following simulations, C will be selected, based on the results from Figs. 5.4 to 5.8, as the limiting values of (3.39), that is, C=1/2 for b=4 and C=1 for b=3, to provide the best compromised tracking error performance.

5.3.1.2 Performance Versus M

In this section, simulation results of the control system for various M values with an ADPCMQ coder when $\rho=10^2$ are presented. As discussed before, C was chosen to be C=1 for b=3 and C=1/2 for b=4. The steady-state performance to a unity set-point are shown in Fig. 5.9 for b=3 and in Fig. 5.10 for b=4 with M varying from 0 to 4. The step-size adaptation range is given by the ratio $R_d=2^{2M}$ or $R_d=12M$ dB.

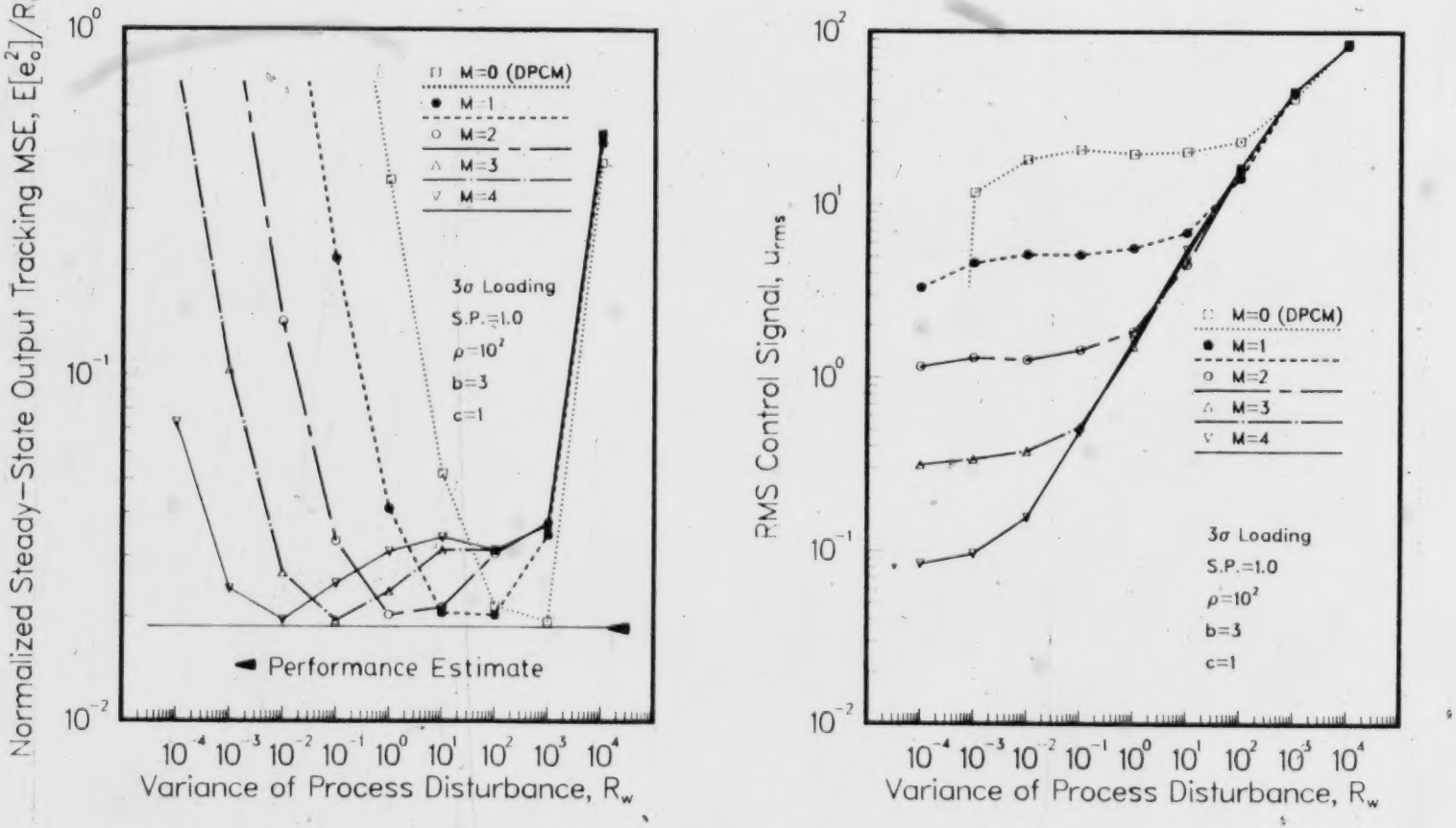


Fig. 5.9 Steady-State Control System Performance for Various *M* with *b*=3 and an ADPCMQ Coder.

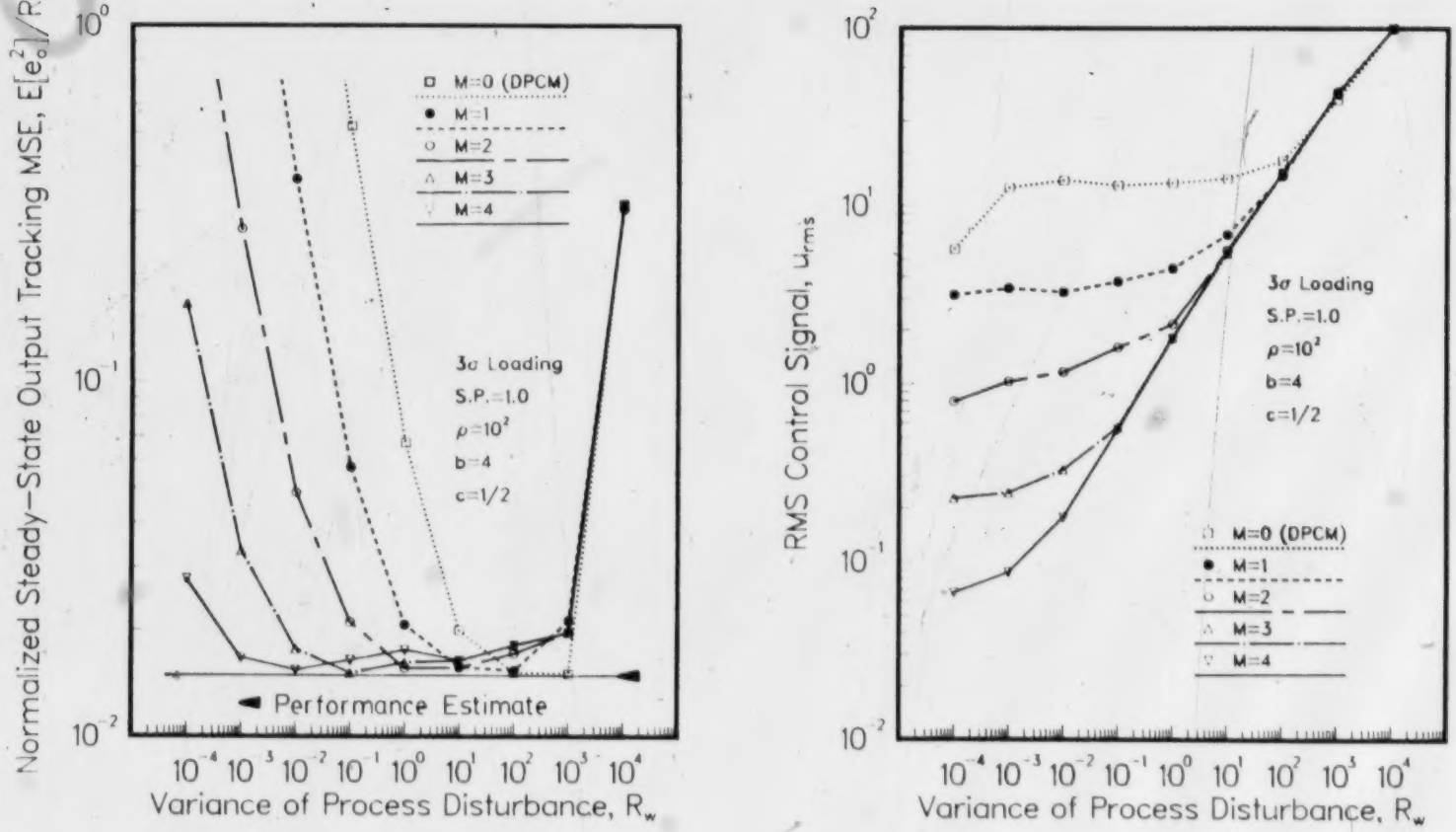


Fig. 5.10 Steady-State Control System Performance for Various *M* with *b*=4 and an ADPCMQ Coder.

When M=0, the ADPCM coder simply reduces to a nonadaptive DPCM coder. With $R_{W_{max}}$ fixed at a specified value, the minimum R_W that can be handled by the quantizer is determined by the range ratio R_a for the different M values. For example, when M=0, $R_{W_{min}}$ is simply $R_{W_{max}}$ because this a nonadaptive DPCM. When M=1, the adaptation range is 12 dB and this will result in an $R_{W_{min}}$ value of about 10^2. As can be observed from Figs. 5.9 and 5.10, a higher M value will generally result in control system performance which is robust to a wider range of R_W. This range, as found from the simulation results, agrees quite well with the range defined by R_a. As discussed before, in all cases, the best performance occurs when R_W is at the minimum value that can be handled by the quantizer. At such an R_W level, the quantizer uses Δ_{min} most of the time and essentially behaves like a fixed quantizer. Hence the performance at this value of R_W agrees fairly well with the performance estimate of (5.10). For R_W other than $R_{W_{min}}$, the performance deviates from the performance estimate. Again, the discrepancy is a result of ignoring the sequential correlation in the quantization noise as well as the effects of quantizer saturation and slope overload effects. This discrepancy is more significant for b=3 than for b=4 as a result of the coarse quantization as discussed before. For b=4, the performance agrees reasonably well with the theoretical performance estimate and M=4 provides, relatively, the widest range of robustness to R_W.

The RMS control signal, u_{rms}, as computed from (5.14) is also shown in Figs. 5.9 and 5.10. The general trend is that when R_w is increasing, it requires more control effort to suppress the effect of the process disturbance. These figures illustrate that for low R_w, the control signal is greatly reduced when M is large. This is a result of the wider range of the quantizer with higher M values. Consequently, better feedback information is available at low R_w levels and hence more effective control signals will result. In the simulations, it was observed that when b=3, M=0, and $R_w=10^{-4}$, the RMS control signal was essentially zero and the system dynamic behaviour was essentially an open-loop response to the process disturbance and the quantization noise. The fixed quantizer in the DPCM coder with a step-size which was too large for this R_w value was unable to detect the error and hence erroneous feedback information was used to generate the control signal. This is a good illustration of the effects on the control system that may be caused by inadequate feedback information.

Besides determining the range of robustness to R_w, M also affects the speed of step-size adaptation. Since the number of available step-sizes is given by $2M+1$ for the algorithms (3.30) and (3.33), it will take a longer time to switch from a smaller step-size to a larger step-size for higher M values. For example, when M=4, it will require at least 9 sampling periods to switch from Δ_{min} to Δ_{max} (or from Δ_{max} to Δ_{min}), but for M=2, this will take only 5

sampling periods. During slope overload, which may introduce large quantization errors to the system, a long delay to switch up from the lower step-sizes to Δ_{max} may have detrimental effects on the control systems. For open-loop unstable systems, this may even lead to instability.

To illustrate effects of the speed of step-size adaptation, with different M values, dynamic responses of the plant output and the step-size to a nonstationary process disturbance are shown in Figs. 5.11 and 5.12 for M=4 and M=2 from k=850 to k=1450 with a zero and non-zero set-point, respectively. Figs. 5.13 and 5.14 are similar to Figs. 5.11 and 5.12 except that the time is from k=3350 to k=3950. In these figures, the process disturbance variance level changes for every 100 sampling periods as indicated by Δ on the figures. It is selected randomly from the set $\{10^{-4}, 10^{-3}, \ldots, 10^{4}\}$ with equal probability. In general, as can be seen from these figures, the step-size adaptation behaviour is determined by the variance of the process disturbance. When M=4, there are more step-sizes from which the quantizer can select to match the R_w level, whereas for M=2, there are less to choose from. Since Δ_{max} is specified with $R_{w_{max}}=10^{3}$ in all cases, Δ_{min} for M=4 is smaller than that of M=2. As a result, the M=4 quantizer is able to adapt its step-sizes to lower R_w levels than that of the M=2 quantizer. But, with M=2, the quantizer is able to switch up from Δ_{min} to Δ_{max}, or vice versa, in a much shorter time than that of M=4.

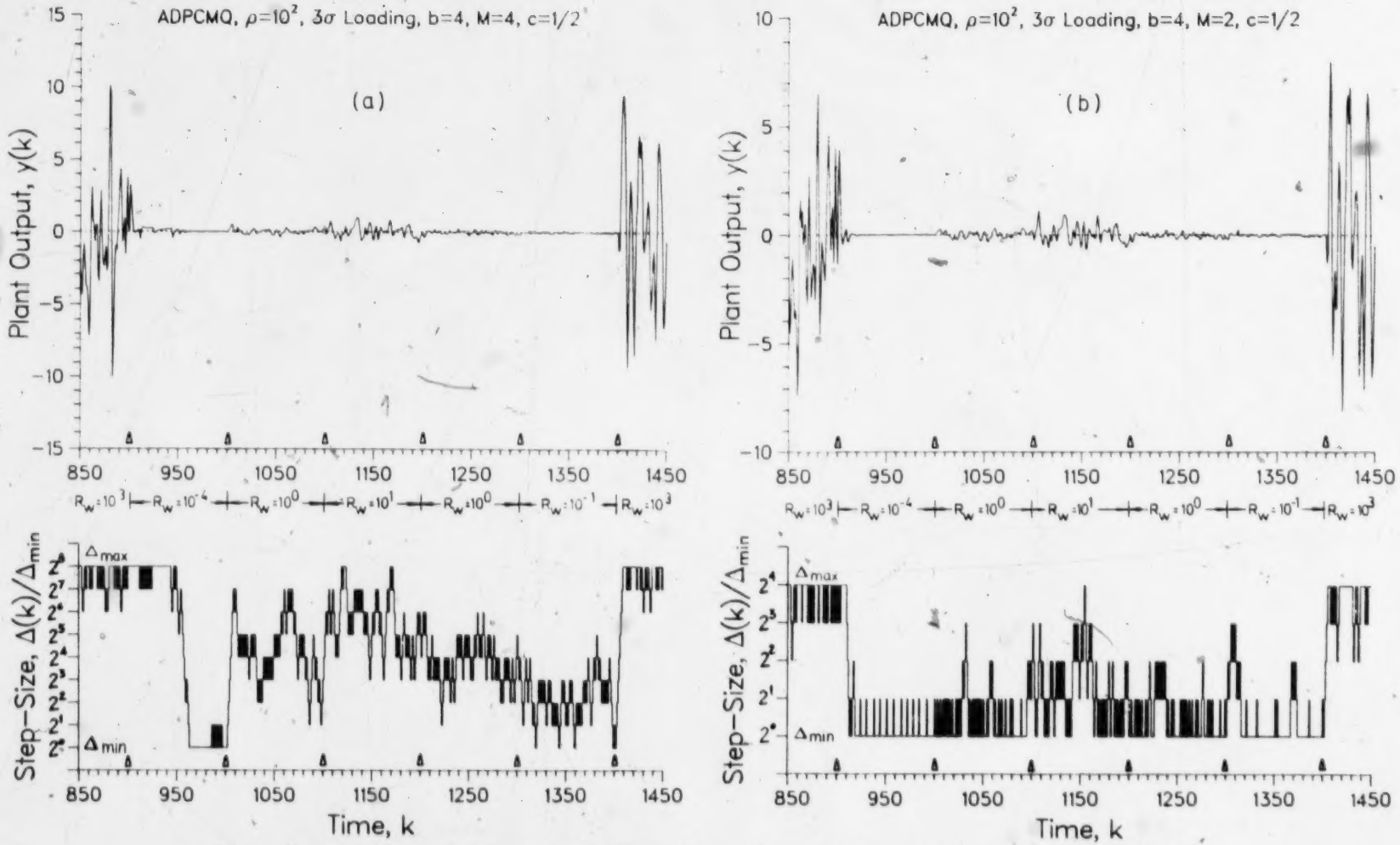


Fig. 5.11 Plant Output and Step-Size Responses to a Nonstationary Process Disturbance and a Zero Set-Point (Regulator) for Time k=850 to k=1450 with an ADPCMQ Coder. (a)M=4. (b)M=2.

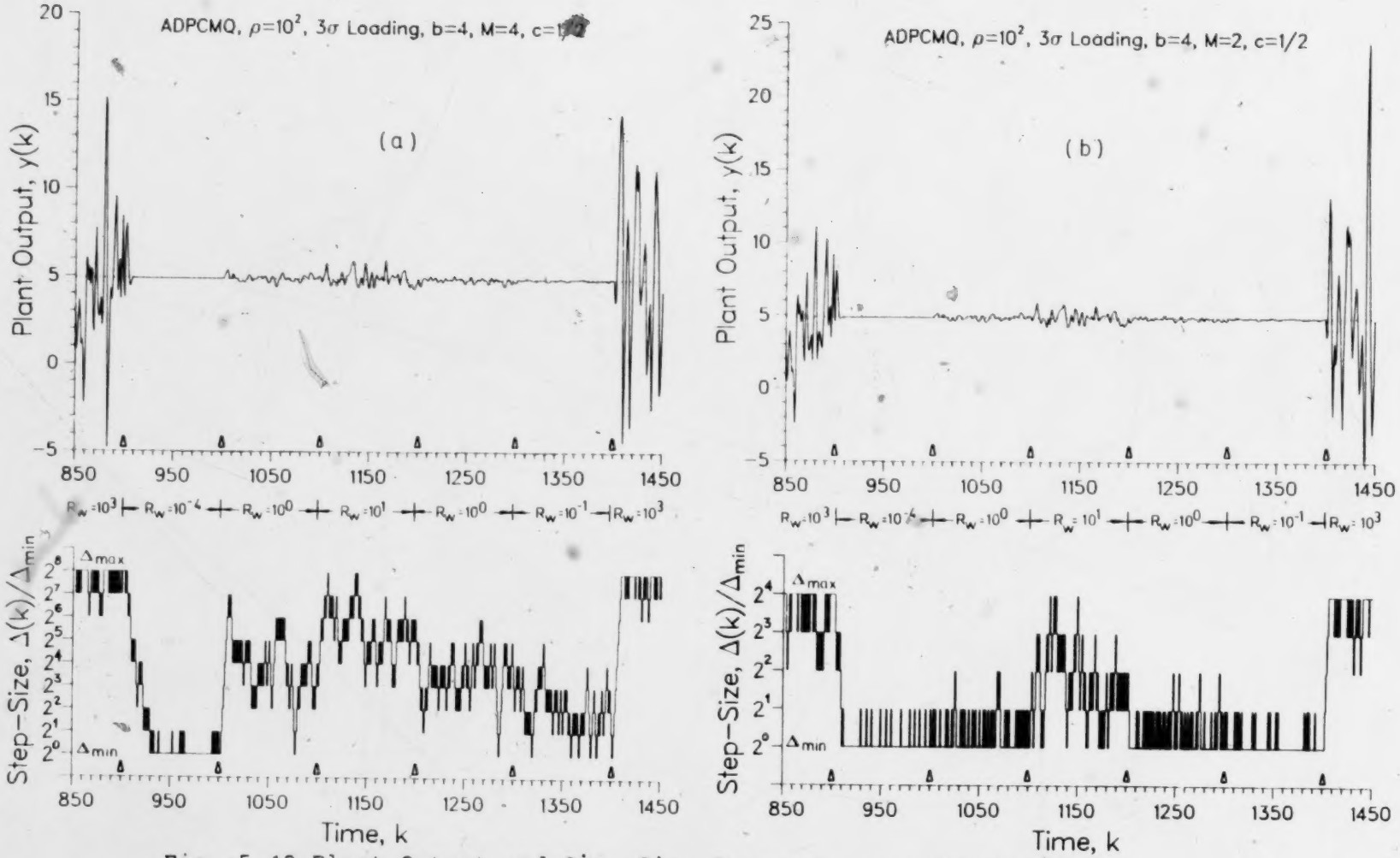


Fig. 5.12 Plant Output and Step-Size Responses to a Nonstationary Process Disturbance and a Constant Non-Zero Set-Point for Time k=850 to k=1450 with an ADPCMQ Coder. (a)M=4. (b)M=2.

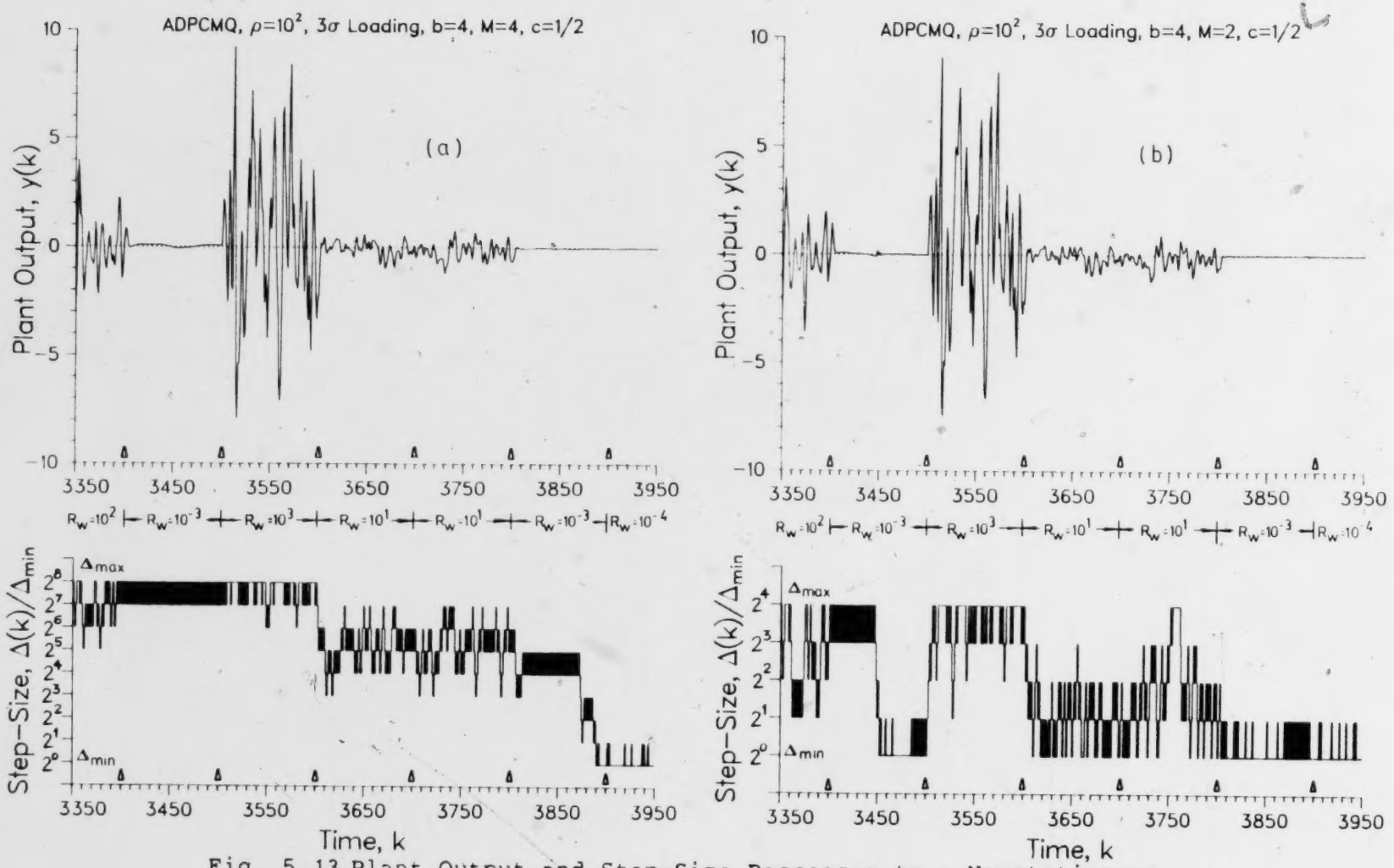


Fig. 5.13 Plant Output and Step-Size Responses to a Nonstationary Process Disturbance and a Zero Set-Point (Regulator) for Time k=3350 to k=3950 with an ADPCMQ Coder. (a)M=4. (b)M=2.

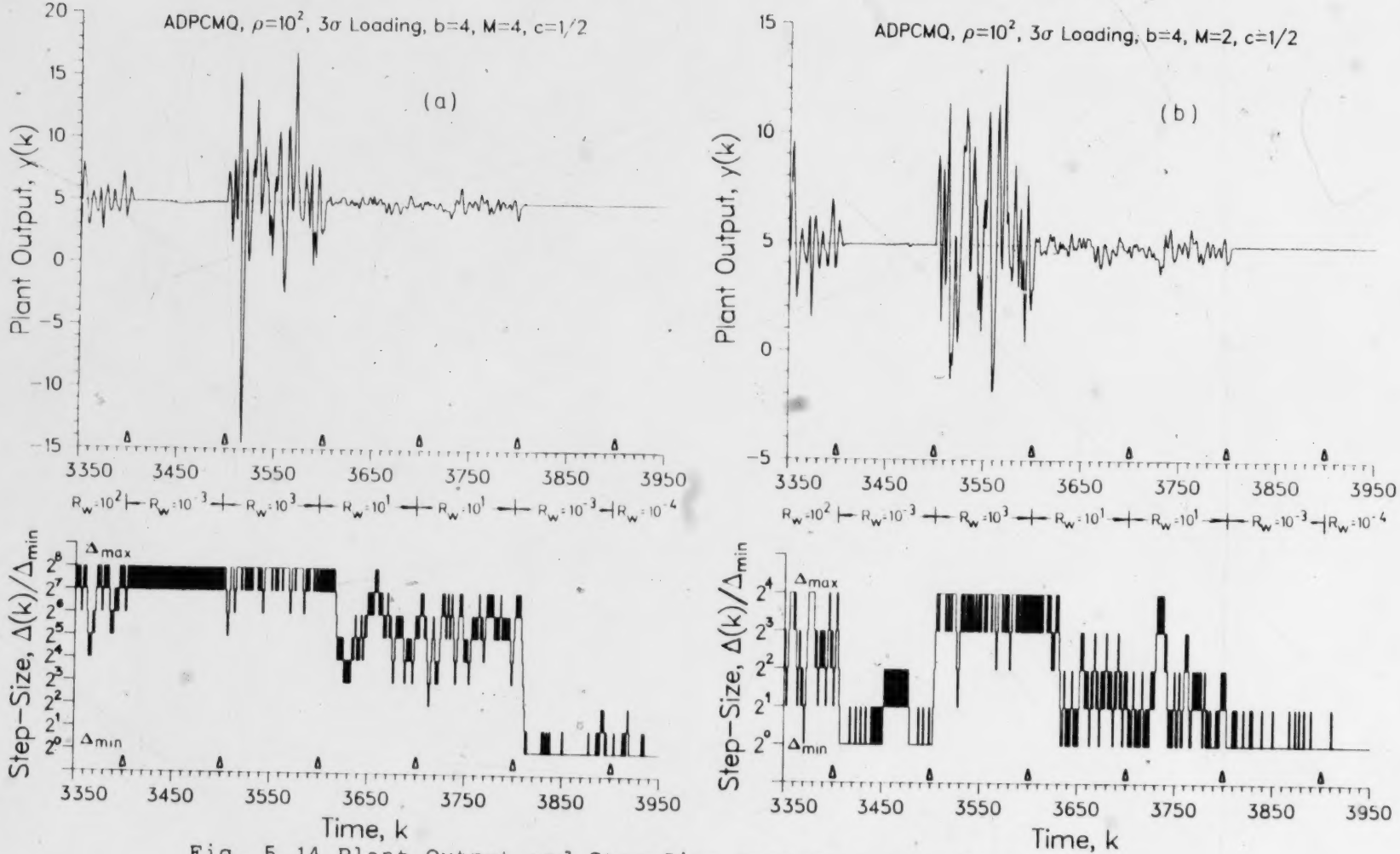


Fig. 5.14 Plant Output and Step-Size Responses to a Nonstationary Process Disturbance and a Constant Non-Zero Set-Point for Time k=3350 to k=3950 with an ADPCMQ Coder. (a)M=4. (b)M=2.

At time k=900, R_w changes from 10^3 to a much smaller value of 10^{-4}. As can be observed from Figs. 5.11 and 5.12, with M=4, it takes a much longer time (about 60 samples and 30 samples, respectively) to switch from the higher step-size (near Δ_{max}) to the more suitable, lower step-sizes (near Δ_{min}) when compared with that required by the system with M=2 (about 10 samples). Furthermore, during the transition from R_w=10^3 to 10^{-4} at k=900, the dynamic behaviour with M=4 for the zero and non-zero set-point situations are quite different. In Fig. 5.11a, during k=900 to about k=940, the regulator system experienced a disturbance, which is reflected from the plant output, caused by a combination of large quantization error and the existence of limit cycles. Consequently, erroneous feedback information is used to generate the control signal which also becomes erroneous and thus results in loss of closed-loop control. Then, after about k=940, the quantizer began to switch down to the lower step-sizes in about 20 samples. In Fig. 5.12a, during the same period of time, such a disturbance did not occur. Starting from about k=900, the quantizer was able to switch down to the lower step-sizes in about 30 samples. Limit cycles were not observed for this case. In contrast to the assumption that δ is independent of the set-point r, the dynamic behaviour of the prediction error and hence the quantizer adaptation behaviour, as observed from this simulation result, can be affected by the externally applied reference input. This illustrates the

inadequacy of linear transfer function approach of analysis which has ignored the effects of the adaptive quantizer nonlinearity. Furthermore, in the analysis and transfer function design, it has been assumed that both w and ϵ are stationary and white as well as mutually uncorrelated. But, in this example, w is assumed to be nonstationary. Morever, ϵ is not necessarily white and is very likely to be correlated to w. All these effects may result in a prediction error δ which is inseparable from w and r. Consequently, this will lead to a possible influence of the set-point on the step-size adaptation as evident from Figs. 5.11 and 5.12.

At time k=3400, R_w changed from 10^2 to a lower level of 10^{-3}. As evident from Figs. 5.13 and 5.14, the M=2 quantizer behaved better than the M=4 quantizer. With M=4, for both the regulator and non set-point situations, the system experienced limit cycle effects from k=3400 to about k=3500. It was observed that limit cycles existed in the control signal u, the plant prediction $\bar{y}$, the feedback signal $\hat{y}$, the prediction error δ, and the step-size Δ. The adaptive quantizer could never switch down to the more suitable lower step-sizes during this period of time. This again illustrates the nonlinear effects of the adaptive quantizer on the control system, which may be detrimental to the system performance. Before R_w changed from 10^2 to 10^{-3} at k=3400, the system might have already been experiencing a large prediction error since R_w was large. This might lead

to a large quantization error. After k=3400 when R_w changed to 10^{-3}, the system was not able to respond to this large jump of R_w quickly enough and hence the prediction error δ as well as the quantization error remained large. Coupled with the sequential correlation in the quantization noise, the adaptive quantizer was not able to respond to this change and continued to behave as if the system were being driven by a large process disturbance. The result was loss of closed-loop control and hence the system output was also affected as can be observed from Figs. 5.13a and 5.14a.

In Fig. 5.13b, during the same period of time, the system also experienced limit cycles similar to that of M=4 but it lasted only for about 50 samples. After about k=3445, the adaptive quantizer was able to switch down to the lower step-sizes and limit cycles no longer existed. In Fig. 5.14b when the set-point is non-zero, during the same period of time, no limit cycles existed and the quantizer was able to switch down to the lower step-sizes after k=3400. This again illustrates the nonlinear effects of the adaptive quantizer, resulting in quantizer behaviour which may be affected by the set-point.

After k=3800 when R_w changed from 10^{1} to 10^{-3}, the system in Fig. 5.13a again experienced limit cycle for about 70 samples before it was able to switch to the lower step-sizes. But for the other systems in Figs. 5.13a, 5.14a, and 5.14b, no limit cycles existed during this period of time.

From the results in Figs. 5.11 to 5.14, one can observe that whenever limit cycles occur, the plant output response will be affected and is generally some sort of an open-loop response as a result of loss of the appropriate control signals. Furthermore, limit cycles in control systems are usually very undesirable since they can cause the control instruments to wear out rapidly and the control energy spent during limit cycles is wasted without producing useful control signals. Basing on the observations from Figs. 5.9 to 5.14, it may be concluded that larger M values are undesirable in applications where the process disturbance may jump abruptly and frequently from a very large variance level to a very small one. Lower M values are more effective in this situation. On the other hand, if a stationary process disturbance is expected, a higher M will be more effective since it can provide a relatively wider range of robustness to R_w. Thus, the best choice of M would be one that yields the best compromise between adaptation speed and the range of robustness to R_w.

5.3.1.3 Performance Comparison of ADPCMQ and ADPCMF Coders

Figs. 5.15 and 5.16 illustrate the control system performance to a unit step reference input with feedforward ADPCM (ADPCMF) and ADPCMQ coders for different quantizer loading factors (3σ and 4σ) with $\rho=10^{-2}$ and $M=4$ using $b=3$ as well as $b=4$ quantizers.

From these figures, it is apparent that ADPCMF coders have a superior performance to that of the ADPCMQ coders,

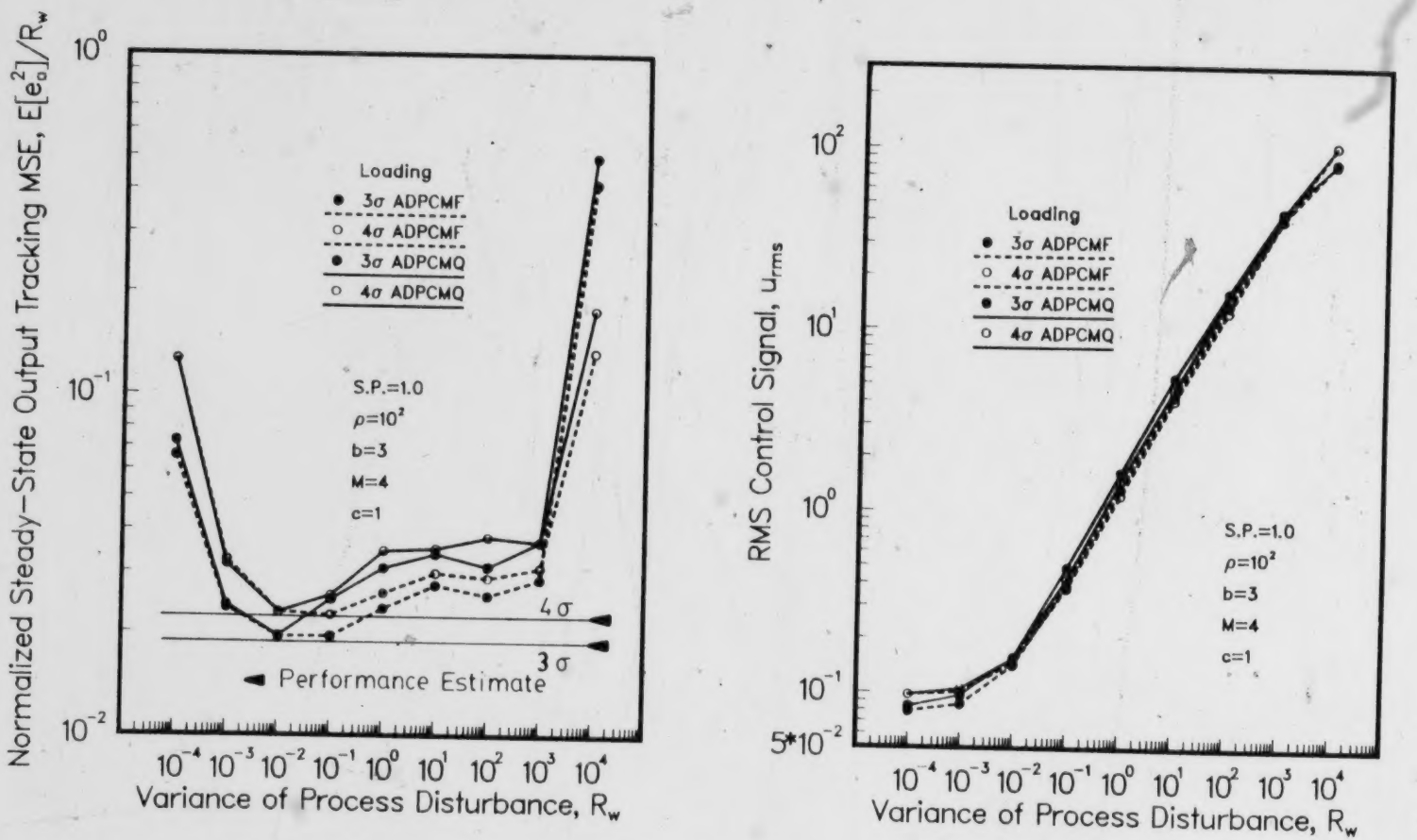


Fig. 5.15 Steady-State Control System Performance Comparison for ADPCMF and ADPCMQ Coders with b=3 and Various Quantizer Loadings.

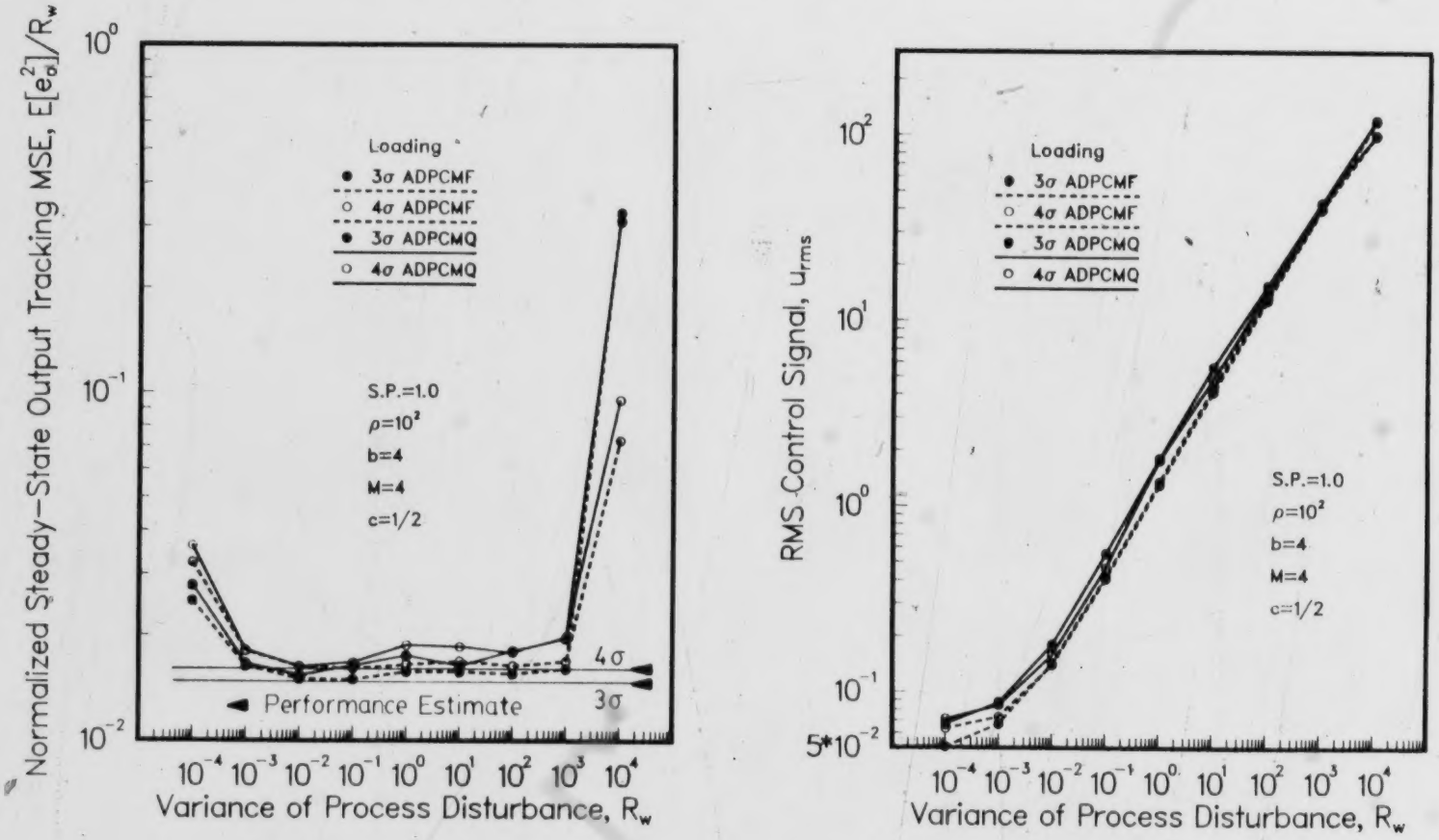


Fig. 5.16 Steady-State Control System Performance Comparison for ADPCMF and ADPCMQ Coders with b=4 and Various Quantizer Loadings.

although the improvement in performance is not very significant except at higher R_w levels. It can also be observed that a more conservative 4σ loaded quantizer will lead slightly inferior performance to that of a 3σ loaded quantizer except for $R_w=10^4$ beyond the assumed maximum value of $R_{w_{max}}=10^3$. The RMS control signals are essentially the same for ADPCMF and ADPCMQ coders as well as for 3σ and 4σ loading.

The improvement in performance of an ADPCMF coder is a result of its structure. Since the most current information (see (3.33)) of the quantizer input is used in the adaptation, the quantizer can make step-size adjustment more rapidly than can the feedback ADPCM coder which uses a delayed (1 sample) version of the quantizer output information (see (3.30)). This is especially important at high R_w levels since by making more up-to-date step-size adjustments, the chances of developing slope overload and quantizer saturation will be reduecd. This benefit is evident from Figs. 5.15 and 5.16 which show that the perfomance improvement is more significant with ADPCMF coders for b=4. For b=3, the ADPCMF coder can improve the performance for higher R_w levels but is still unable to avoid the results of the coarse quantization step-size effect and hence the performance still deviates from the performance estimate. For b=4, however, the performance of the control system with ADPCMF agrees quite well with the performance estimate over a wide dynamic range of R_w levels

as given by the ratio R_d.

Dynamic responses to a nonstationary process disturbance similar to those of Figs. 5.11 to 5.14 were also simulated and the results are shown in Figs. 5.17 and 5.18 for zero and non-zero set-points, respectively, for an M=4, b=4, and 3σ loaded quantizer. As apparent from these figures, the undesirable limit cycle effects that occurred in the previous cases did not exist in the ADPCMF systems for both the regulator and non-zero set-point responses from k=900 to k=1000, from k=3400 to k=3500, and from k=3800 to k=3900. This illustrates the benefit of using the current quantizer input information for step-size adaptation rather than the one-sample delay quantized information.

In general, ADPCMF coder can provide better performance than can an ADPCMQ coder. However, the implementation of an ADPCMF coder requires extra complex hardware as compared with the simpler ADPCMQ coder hardware requirement. Therefore, before deciding which coder to be used, one must consider the trade-off between system performance and the hardware complexity. Nevertheless, in light of the ability to avoid limit cycle effects, ADPCMF coders may be preferrable in applications where a nonstationary process disturbance is to be expected.

5.3.2 Performance Comparisons of ADPCM Coders and Fixed A/D Converters

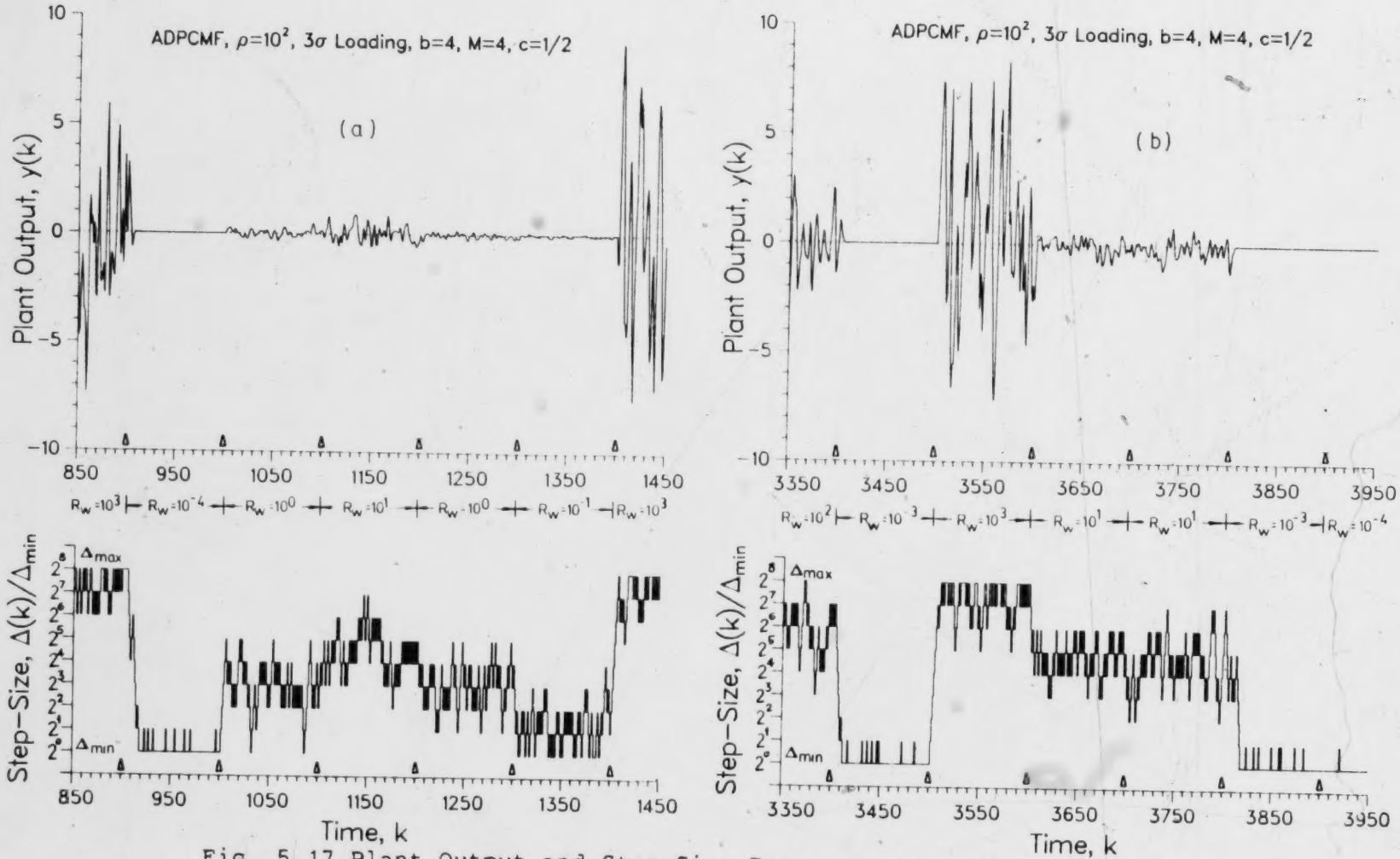


Fig. 5.17 Plant Output and Step-Size Responses to a Nonstationary Process Disturbance and a Zero Set-Point (Regulator) with an ADPCMF Coder. (a)Time k=850 to k=1450. (b)Time k=3350 to k=3950.

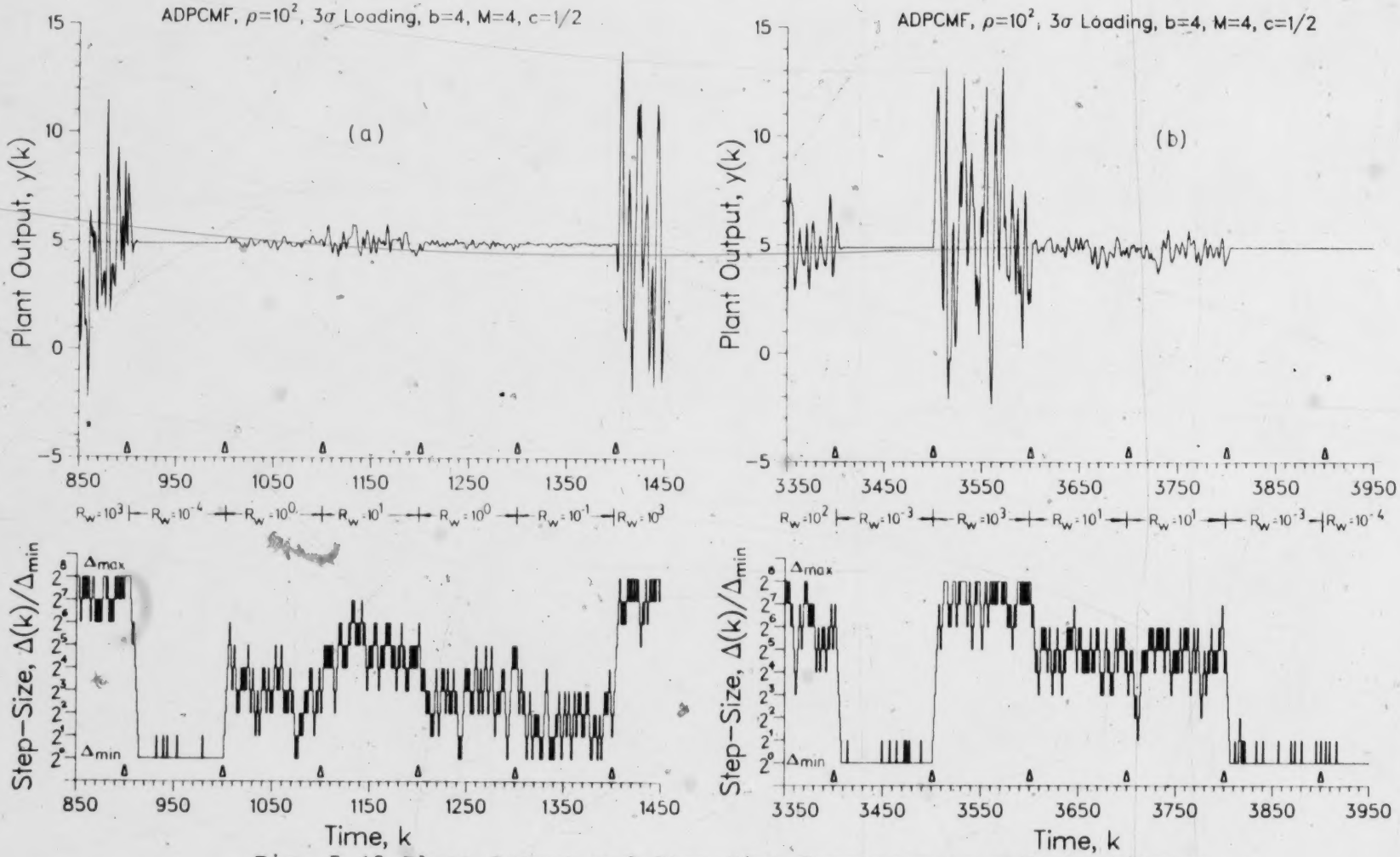


Fig. 5.18 Plant Output and Step-Size Responses to a Nonstationary Process Disturbance and a Constant Non-Zero Set-Point with an ADPCMF Coder. (a)Time k=850 to k=1450. (b)Time k=3350 to k=3950.

This section presents comparisons of control performance of systems with an ADPCM coder with that of systems with a fixed A/D converter having the same number of bits (b=4). As in many conventional digital control configurations, the fixed A/D converter is placed in the forward path of a unity feedback control system. Therefore, instead of quantizing the actual plant output feedback signal, the tracking error at the reference input will be quantized and coded into digital information to be applied to the controller $D(z)$.

With the fixed A/D in the forward path, it can be shown that the steady-state MSE of the tracking error at the reference input is given by the same equation (5.7) which is the output tracking MSE when a fixed quantizer is in the feedback path of the control system. Normalizing this equation as before, one can then obtain the same form of normalized MSE as given by (5.10). This information can then be used in scaling the fixed quantizer and is readily obtainable by specifying the Kalman filter gain vector **L** and the LQG state feedback gain vector **K** and computing (5.10) numerically as before in the ADPCM system. Since a comparison between the ADPCM and the fixed A/D systems is needed, therefore, the same parameters R_v, **K**, and **L** will be used in calculating the fixed A/D scaling. In the simulation, a conservative 3σ loading factor is chosen for a b=4 quantizer. The full scale range, V_{FS}, of the fixed quantizer is selected to be the maximum of

$\max\{r_{max}, 3\sqrt{R_{W_{max}} \cdot [E[e_o^2]/R_W]}\}$ with $E[e_o^2]/R_W$ obtained from Fig. 5.3 and r_{max} being the maximum set-point value. $R_{W_{max}}$ is chosen to be the same as before in the ADPCM system. r_{max} is selected to be 10 in the simulations. The step-size can then be determined from $\Delta = V_{FS}/2^{b-1}$ once V_{FS} is determined.

5.3.2.1 Steady-State Tracking Performance

The steady-state tracking error performance are shown in Figs. 5.19 to 5.24 for various values of ρ. In the simulations, for $\rho=1$ to 10^2, V_{FS} is given by $3\sqrt{R_{W_{max}} \cdot [E[e_o^2]/R_W]}$ and for $\rho=10^3$ to 10^4, V_{FS} is given by r_{max}. Thus, for the latter cases, the quantizer scaling is more conservative than that of the former cases. The results of Figs. 5.19 to 5.24 suggest that the ADPCMF coder generally yields the best control system performance in terms of robustness to R_W and good agreement to the performance estimate (5.10). The figures also show that systems with ADPCMF coders require, relatively, the least amount of RMS control efforts. The performance of ADPCMQ coders is only slightly inferior to that of ADPCMF coders except when $\rho=10^3$ and 10^4 where the difference in performance is more significant. The fixed A/D converter can match to the performance of the ADPCM coders only at $R_W=10^3$, which is the designed value for the quantizer scaling factor. For R_W other than $R_{W_{max}}$, the fixed A/D converter is not able to maintain a robust control performance to the different process disturbance variances. Furthermore, a higher RMS control signal is required when compared with

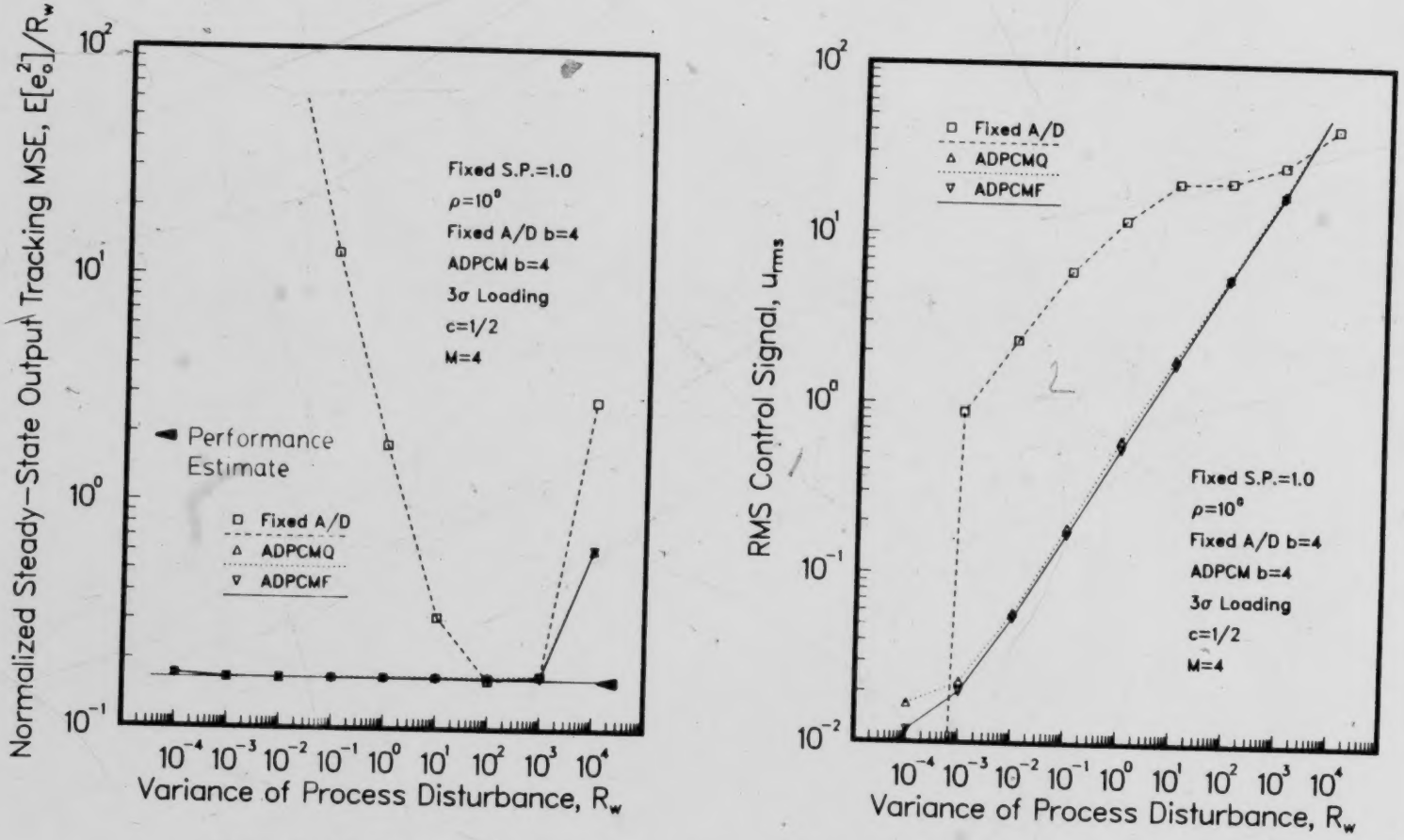


Fig. 5.19 Steady-State Control System Performance Comparison for Fixed A/D, ADPCMQ, and ADPCMF for $\rho=1$.

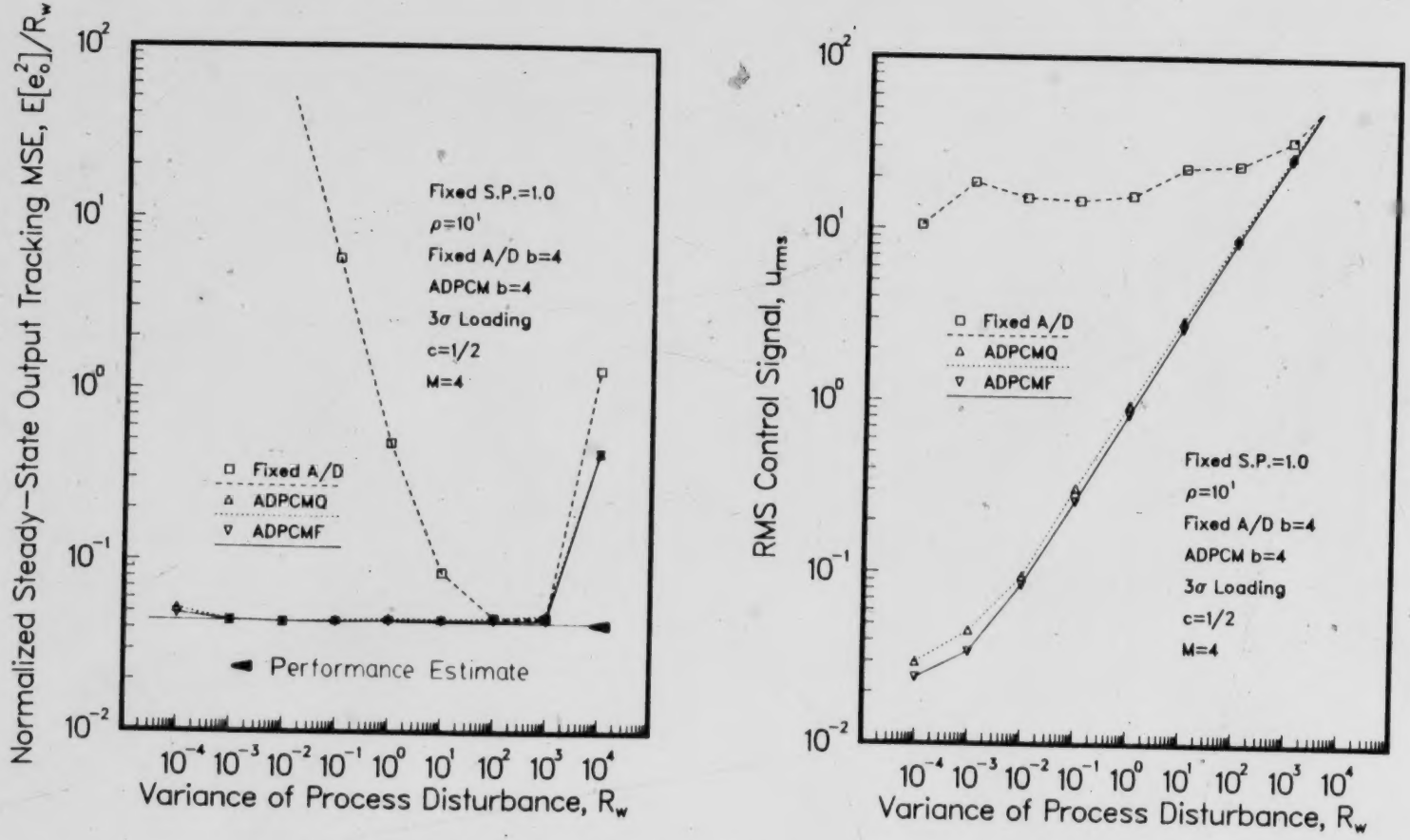


Fig. 5.20 Steady-State Control System Performance Comparison for Fixed A/D, ADPCMQ, and ADPCMF for $\rho=10$.

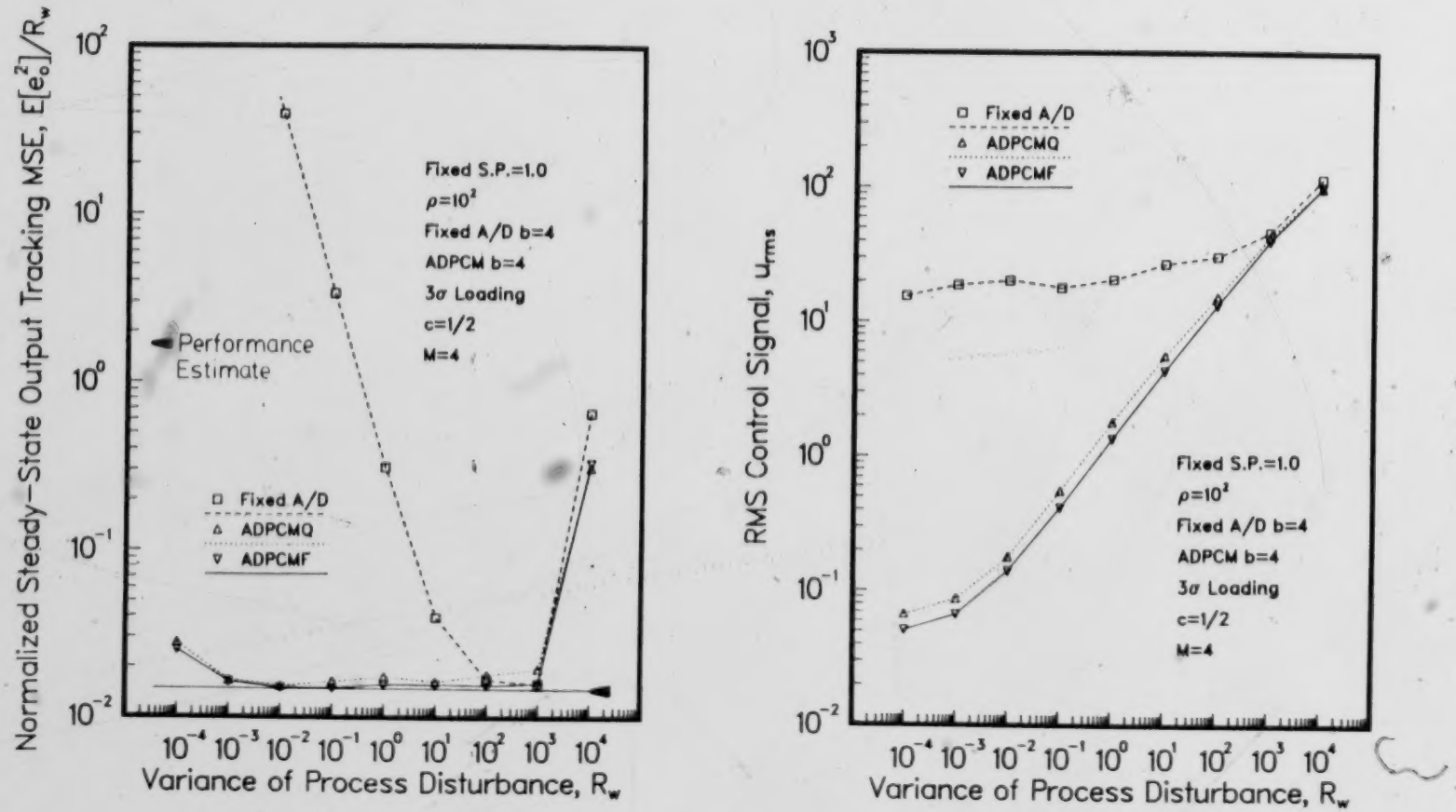


Fig. 5.21 Steady-State Control System Performance Comparison for Fixed A/D, ADPCMQ, and ADPCMF for $\rho=10^2$.

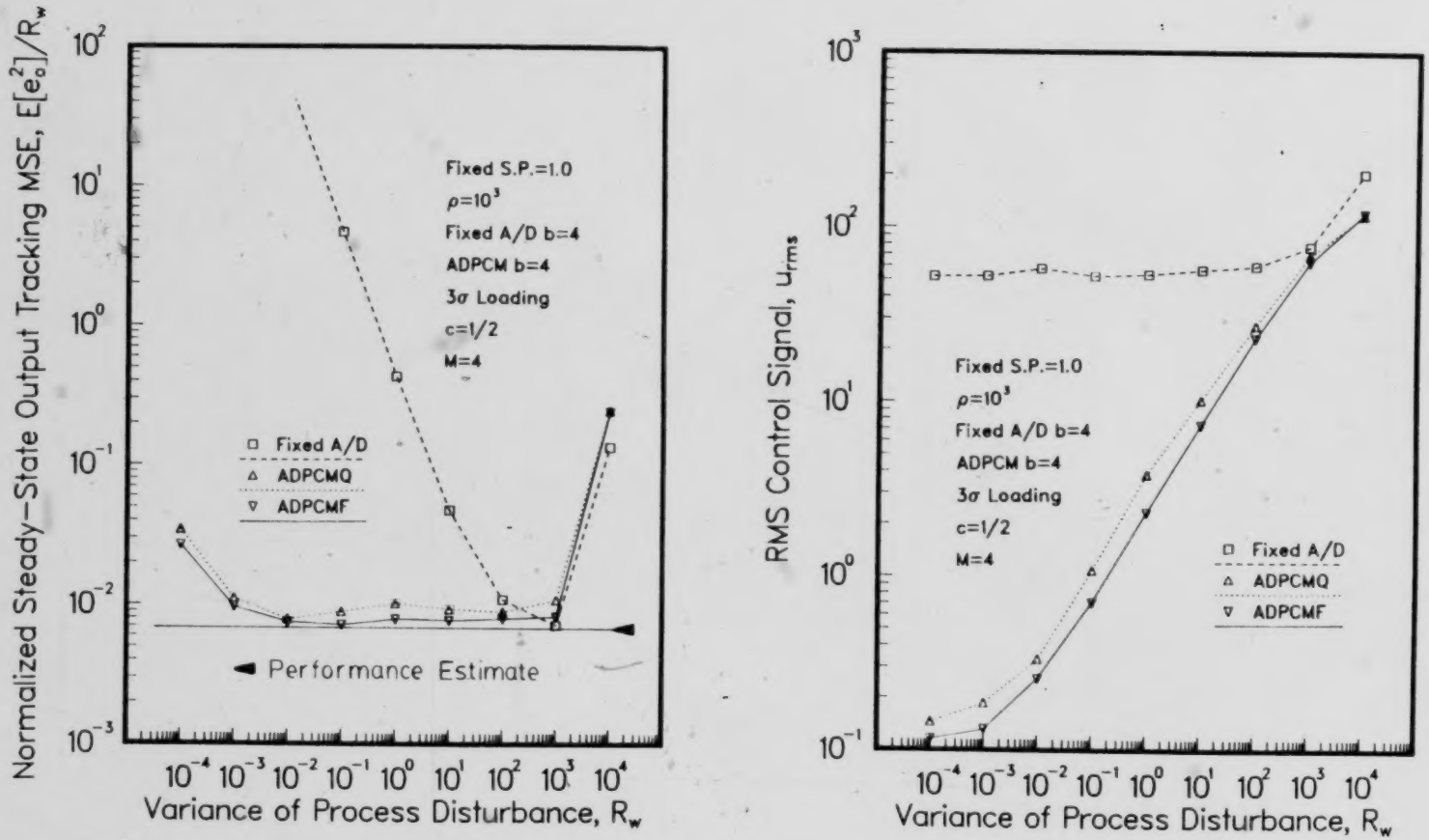


Fig. 5.22 Steady-State Control System Performance Comparison for Fixed A/D, ADPCMQ, and ADPCMF for $\rho=10^3$.

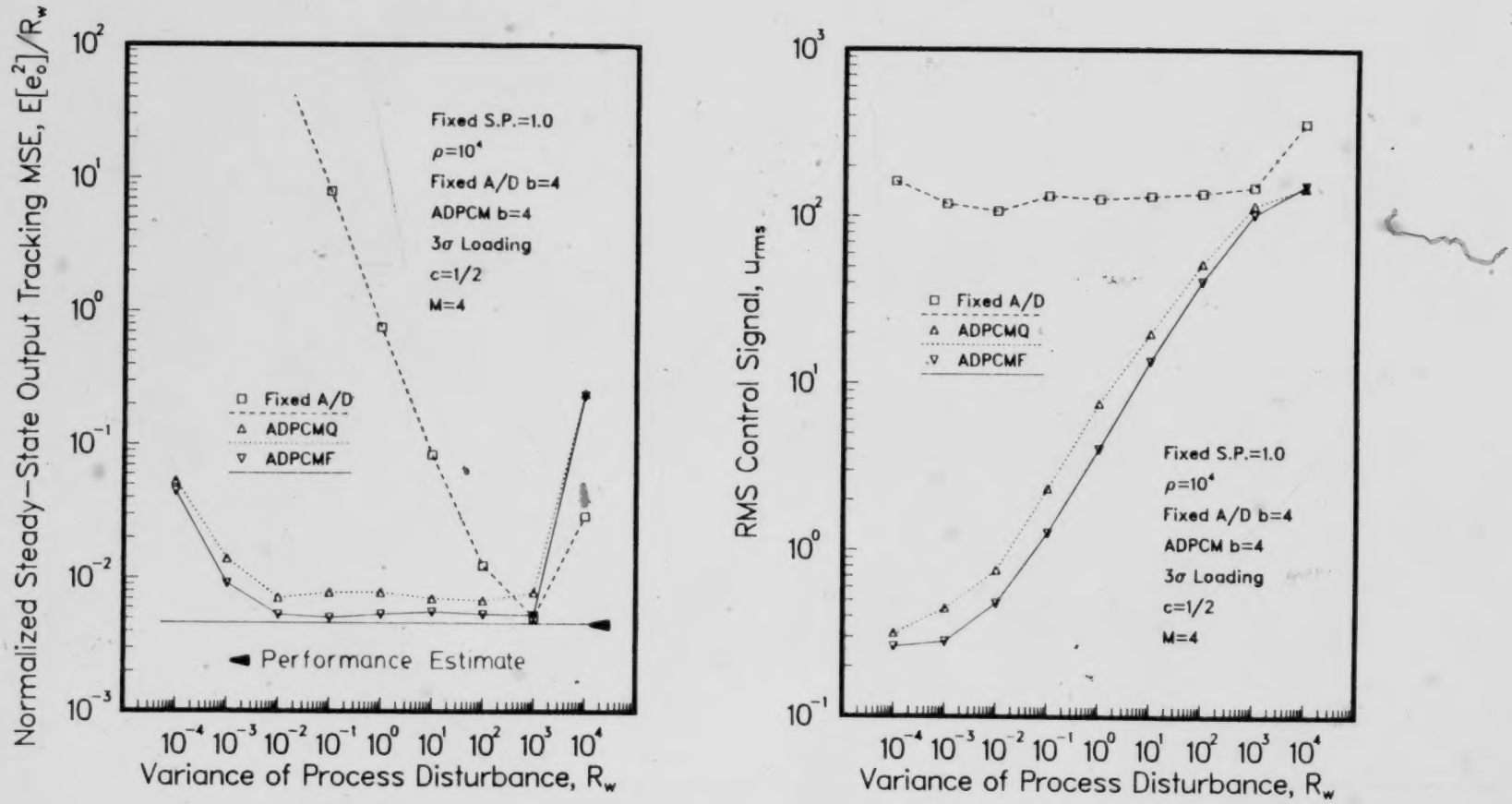


Fig. 5.23 Steady-State Control System Performance Comparison for Fixed A/D, ADPCMQ, and ADPCMF for $\rho=10^4$.

that for the ADPCM coders. Therefore the total cost, or the loss function, for the control system will be higher with a fixed quantizer. This shows the advantage of an ADPCM coder which is capable of reducing the control signal, especially for low R_W levels.

For ρ=1, at R_W=10^{-4}, the control signal is essentially zero for the fixed A/D converter system. The plant output is essentially an open-loop response to the process disturbance. This is a result of the loss of feedback information when the quantizer step-size is too large for the tracking error, at this low R_W value, to be detected by the A/D converter. As a result, erroneous control signal is generated from the controller and thus results in an essentially open-loop response. This situation always causes limit cycles with open-loop unstable plants (Moroney, 1983). Hence it demonstrates that a fixed A/D converter which has been scaled for a particular R_W level will be very ineffective when it is applied to other R_W levels.

For low LQG weightings (ρ=1 to 10^2), the control system performance for ADPCM coders agree reasonably well with the performance estimate for a wide range of R_W levels as specified by R_e (about 50 dB for M=4). The RMS control signal is much smaller than that required by the fixed A/D system for low R_W levels. Even at R_W=10^4, which is beyond the maximum allowable value of $R_{W_{max}}$=10^3, the ADPCM performance is superior to that of the fixed A/D converter.

For higher LQG weightings ($\rho=10^3$, and 10^4), the control system performance of the ADPCMF coder still agrees reasonably well with the performance estimate within the specified range of R_w. For the ADPCMQ coder, however, the performance deteriorates significantly. This is because of the high gain of the LQG compensator when ρ is very high. As can be seen from Fig. 5.3, as ρ is increasing, the gain G_c is also increasing. This indicates that as the LQG weighting is increasing, the control system becomes more sensitive to the quantization error. The errors introduced by quantizer saturation and slope overload, which occur more often in ADPCMQ systems than in ADPCMF systems, will then have a significant effect on the control system performance when ρ is getting higher. Another possible cause of this phenomenon is the limit cycle effects which have been observed in ADPCMQ coders but not in ADPCMF coders (see Figs. 5.13 and 5.14). When the system is driven into limit cycles, the control system performance will then deteriorate significantly since it is essentially an open-loop response.

In all cases in Figs. 5.19 to 5.23, ADPCMF performs better than ADPCMQ and requires less amount of control effort in the control system. Hence, the total cost for systems with ADPCMF coders will be lower than the cost for systems with ADPCMQ or fixed A/D converters.

Finally, the steady-state perfomances are summarized in Figs. 5.24, 5.25, and 5.26 for fixed A/D converter, ADPCMQ coder, and ADPCMF coder, respectively, for the various ρ

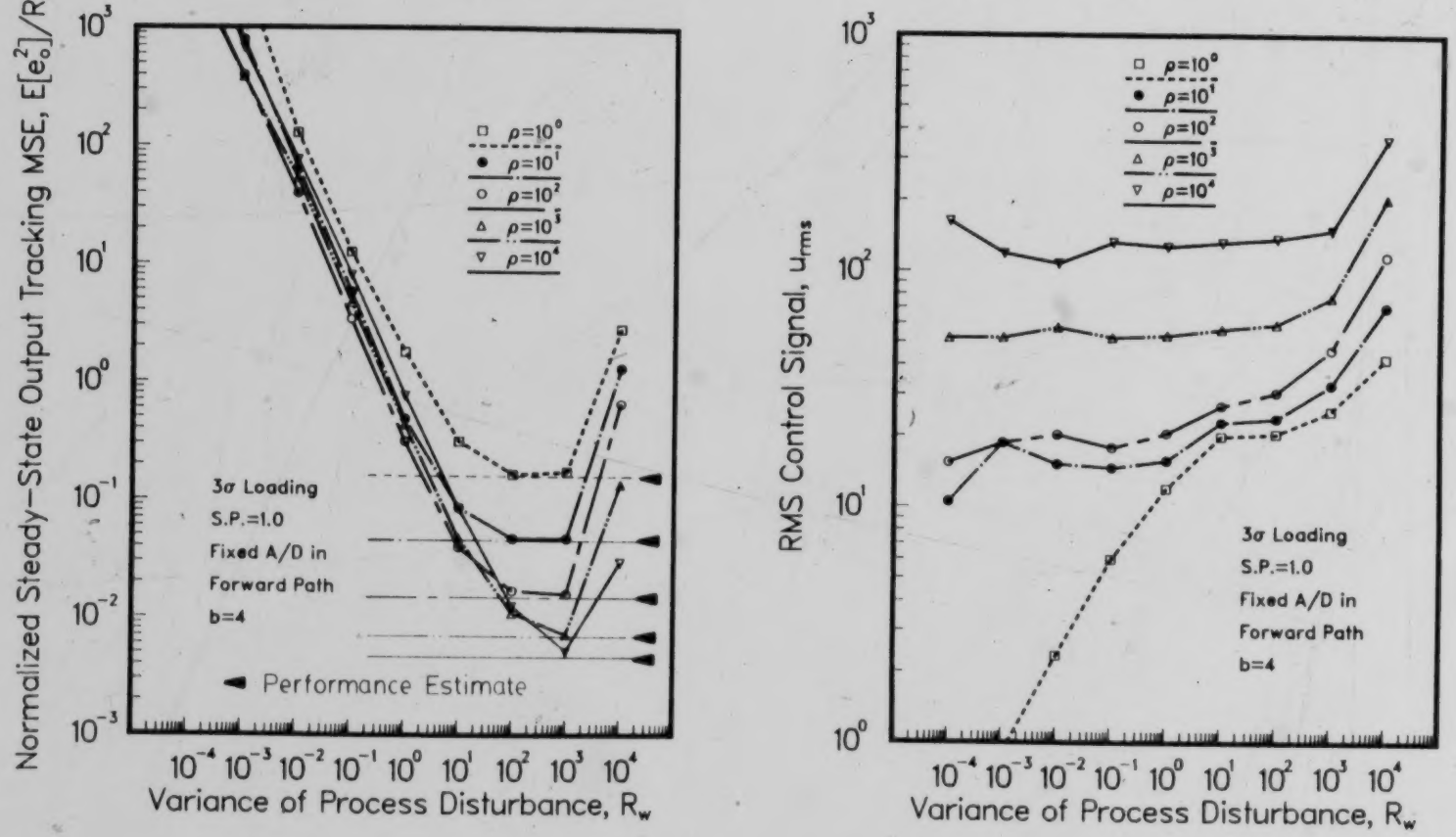


Fig. 5.24 Steady-State Control System Performance of Fixed A/D for b=4 and Various LQG Weighting Factors ρ.

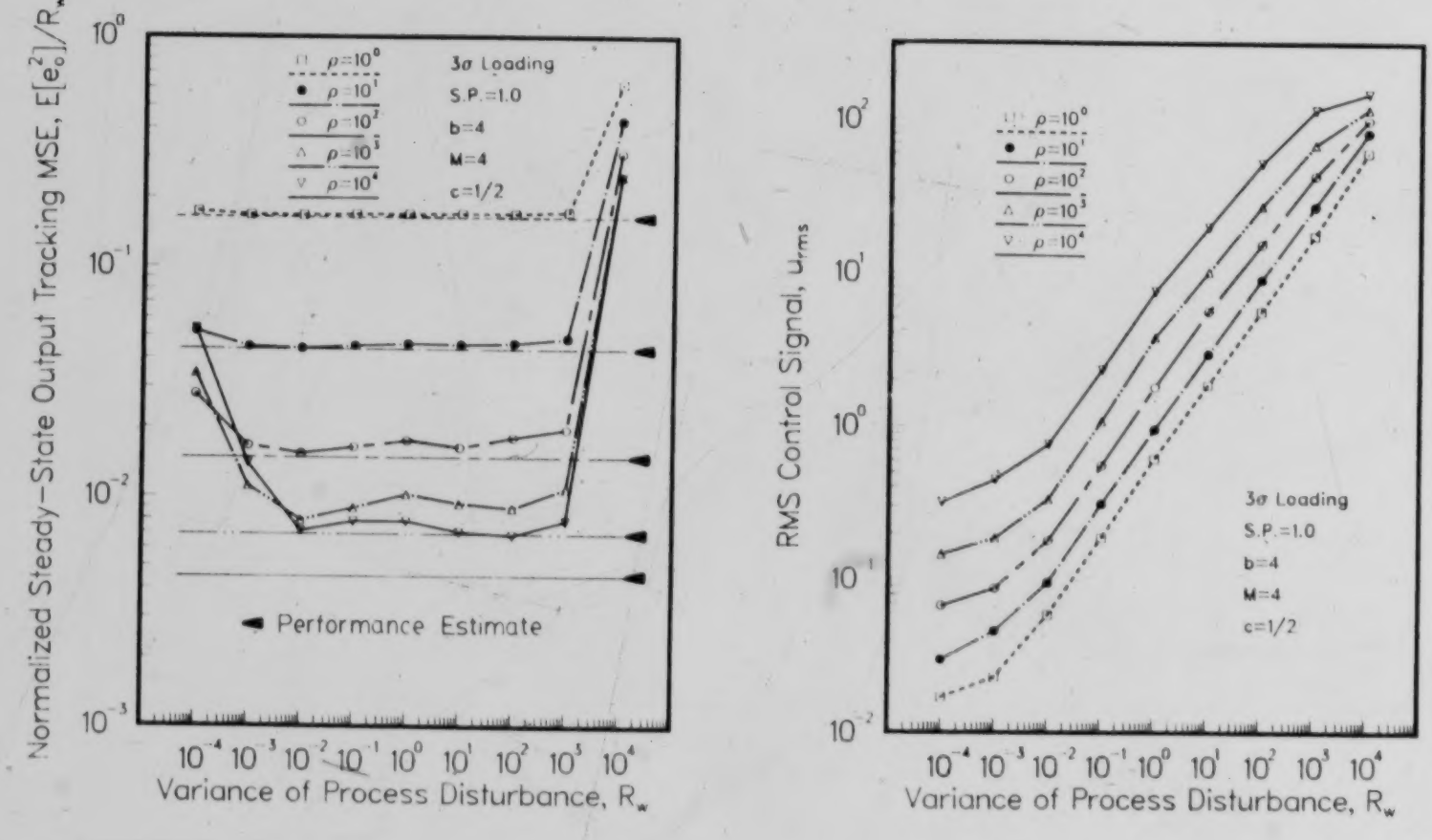


Fig. 5.25 Steady-State Control System Performance of ADPCMQ Coder for b=4 and Various LQG Weighting Factors ρ.

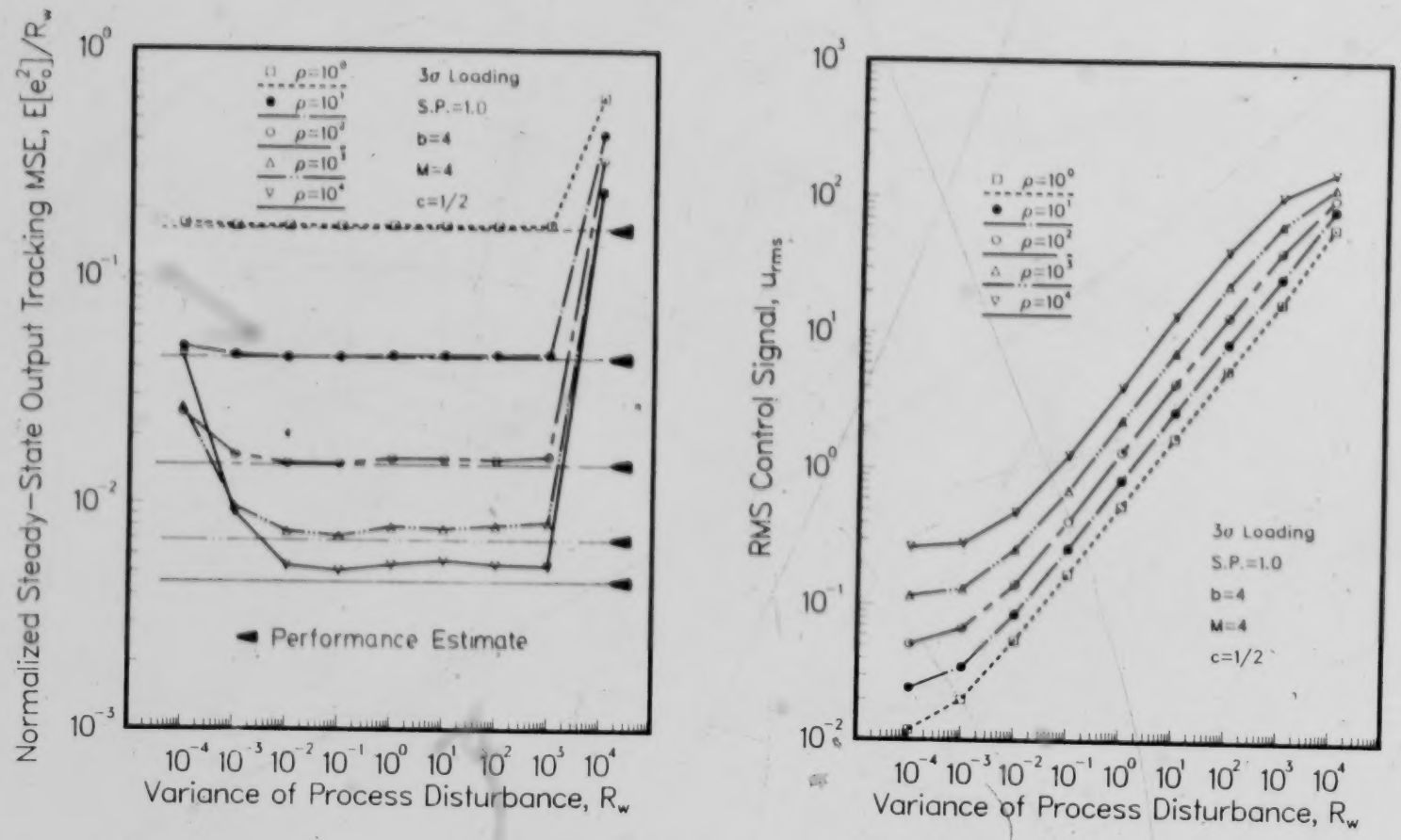


Fig. 5.26 Steady-State Control System Performance of ADPCMF Coder for b=4 and Various LQG Weighting Factors ρ.

factors. As can be observed from these figures, the required steady-state RMS control efforts also increase for increasing R_w, indicating that more control energy is required to suppress the more noisy and exciting process disturbance. On the contrary, the control signal required by the fixed A/D system is relatively insensitive to the R_w levels. This illustrates the ineffectiveness of a fixed A/D converter which has been scaled to a particular R_w value but used in a wide range of R_w levels. The resulting total cost will be unnecessarily high, especially at low R_w levels.

Comparing the steady-state tracking error performance, it is generally true that with higher ρ factors, the MSE will have a lower value. For the fixed A/D converter, the lowest MSE value for each ρ agrees quite well with the performance estimate value only at $R_w=10^3$ which is the designed scaling value. For R_w other than this maximum value, the performance will deteriorate. For the ADPCM coders, the performance agrees reasonably well with the performance estimate except for ADPCMQ coder at high ρ values. This range of robustness also agrees well with that limited by R_d. Thus, in general, it can be concluded that an ADPCM coder performs much better than a fixed A/D converter having the same number of quantizer bits b.

5.3.2.2 Transient Performance Comparisons

This section presents simulation results of the long-term mean square tracking error as computed from (5.13) when the reference input is a series of random set-point

changes for fixed A/D converters and ADPCM coders. For every 100 samples, the set-point is randomly selected from a uniform random number generator distributed in the interval $(0, r_{max})$ with $r_{max}=10$.

Figs. 5.27 to 5.29 illustrate the simulated dynamic responses of the control system with random set-point changes and a stationary process disturbance. The time at which the set-point is changed is indicated by ▲ on the diagrams. Comparing these figures, it is evident that the behaviour of the adaptive quantizer is not affected by the reference set-point values. Instead, the step-size adaptation behaviour is determined by the power level of the process disturbance. Fig. 5.27 illustrates that the median step-size, selected by the quantizer when $R_W=10^{-1}$, has a value of about $2^2\Delta_{min}$. For $R_W=1$, Fig. 5.28 suggests a median step-size value which is between $2^4\Delta_{min}$ and $2^3\Delta_{min}$. Fig. 5.29 suggests a median value which is between $2^4\Delta_{min}$ and $2^5\Delta_{min}$ when $R_W=10$. Thus it can be seen that the adaptive quantizer is capable of seeking a step-size which is the most suitable for the current R_W level. Figs. 5.30 to 5.34 present the output tracking MSE, for $b=4$ and various values of ρ factors, when the control system is subject to random step inputs and a stationary process disturance. This results in a performance indicator which also includes the transient effects. Comparing these figures, it can be seen that the ADPCM coders perform significantly better than the fixed A/D converter in general. There is also a significant

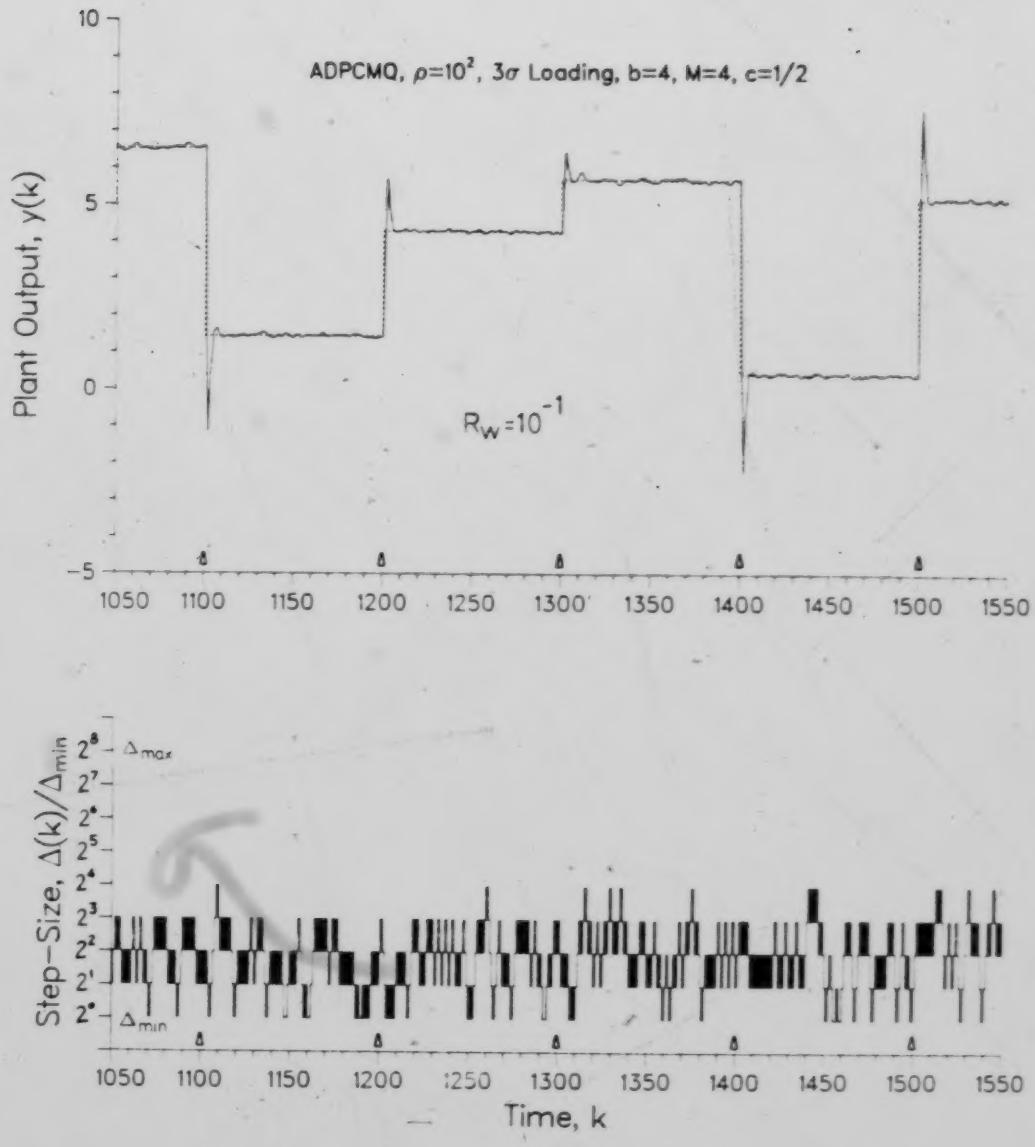


Fig. 5.27 Plant Output and Step-Size Responses to Random Set-Points for $R_w=10^{-1}$.

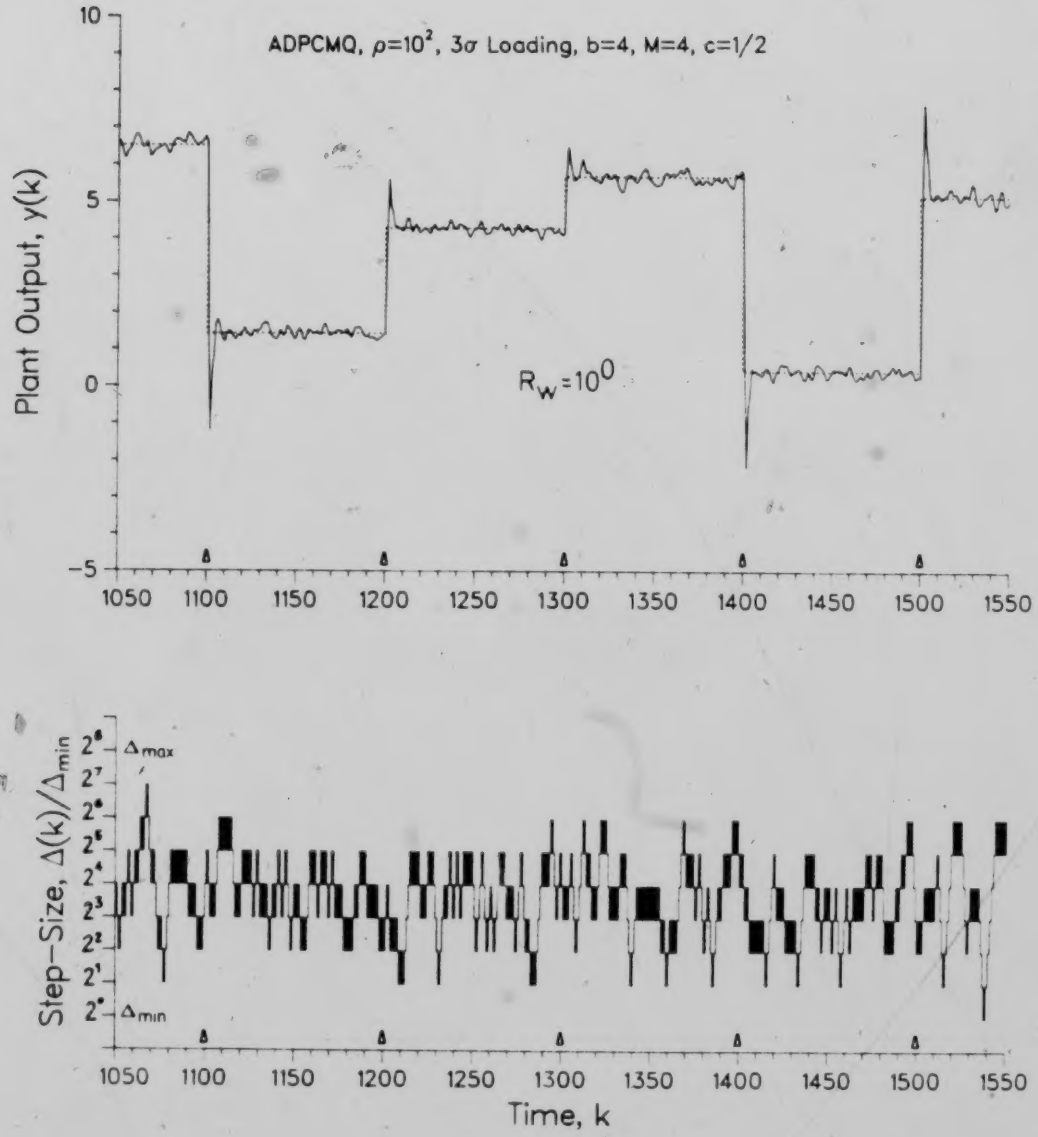


Fig. 5.28 Plant Output and Step-Size Responses to Random Set-Points for $R_W=1$.

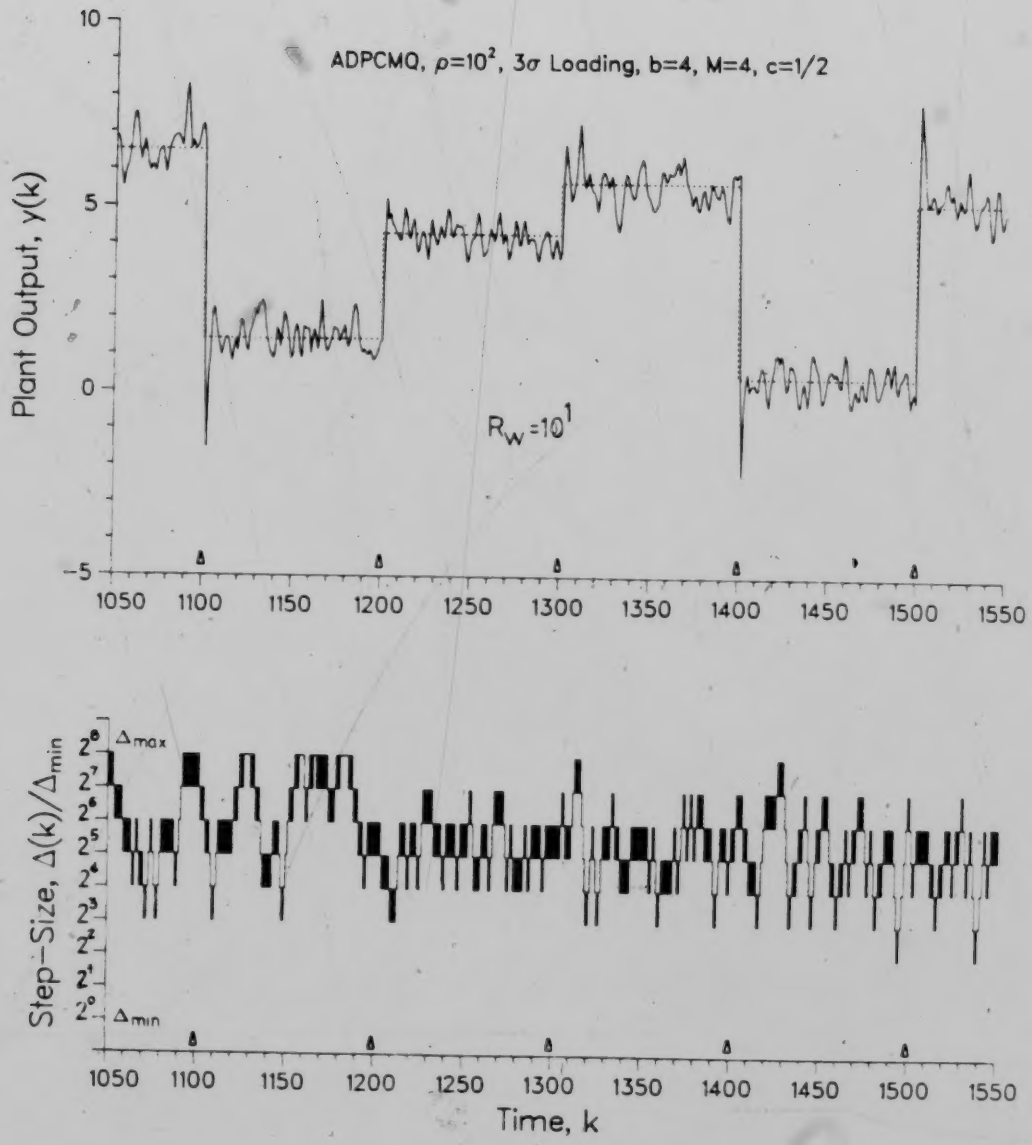


Fig. 5.29 Plant Output and Step-Size Responses to Random Set-Points for $R_w=10$.

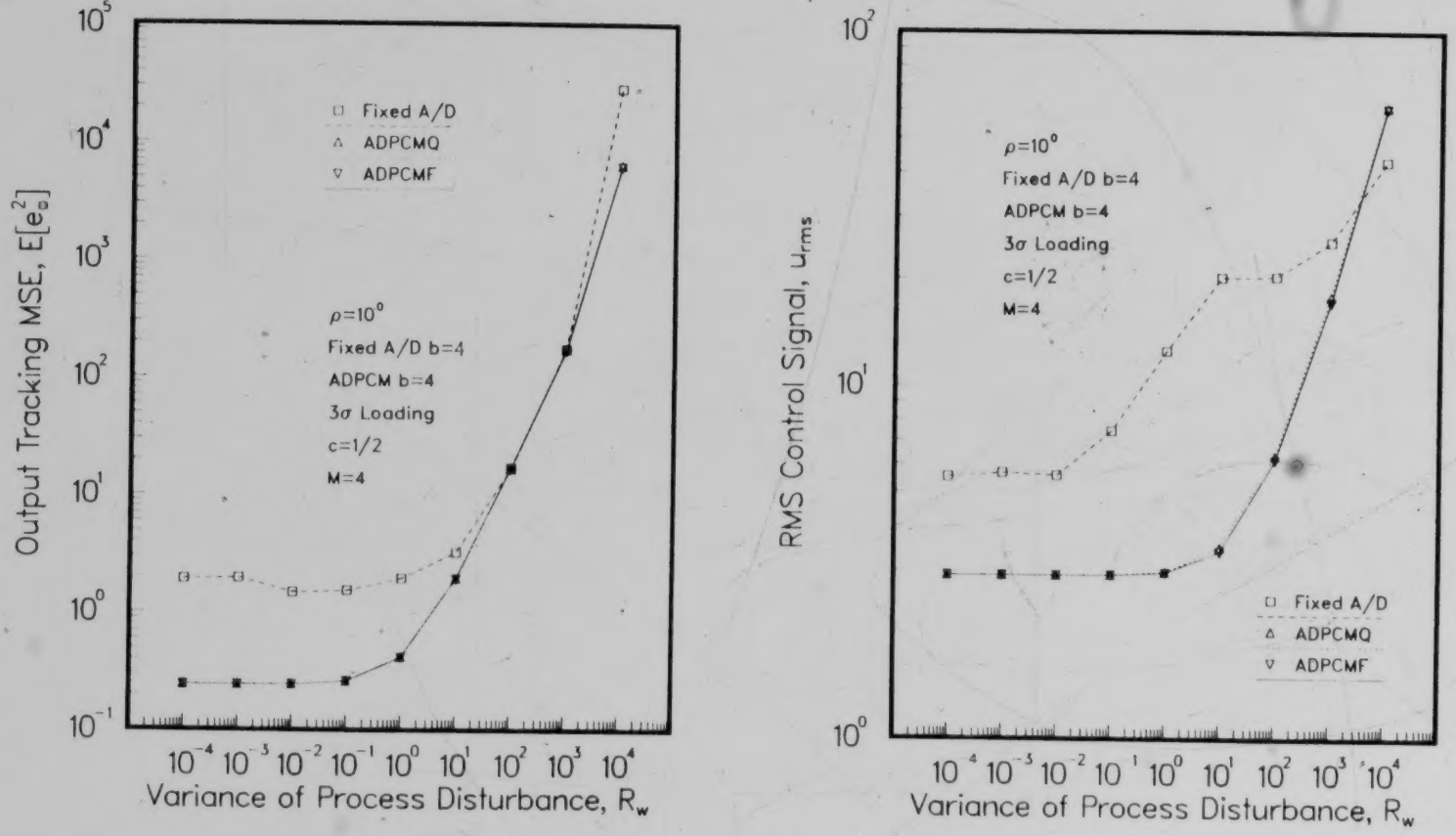


Fig. 5.30 Control System Performance Comparison for Fixed A/D, ADPCMQ, and ADPCMF Coders with ρ=1 and Random Set-Points.

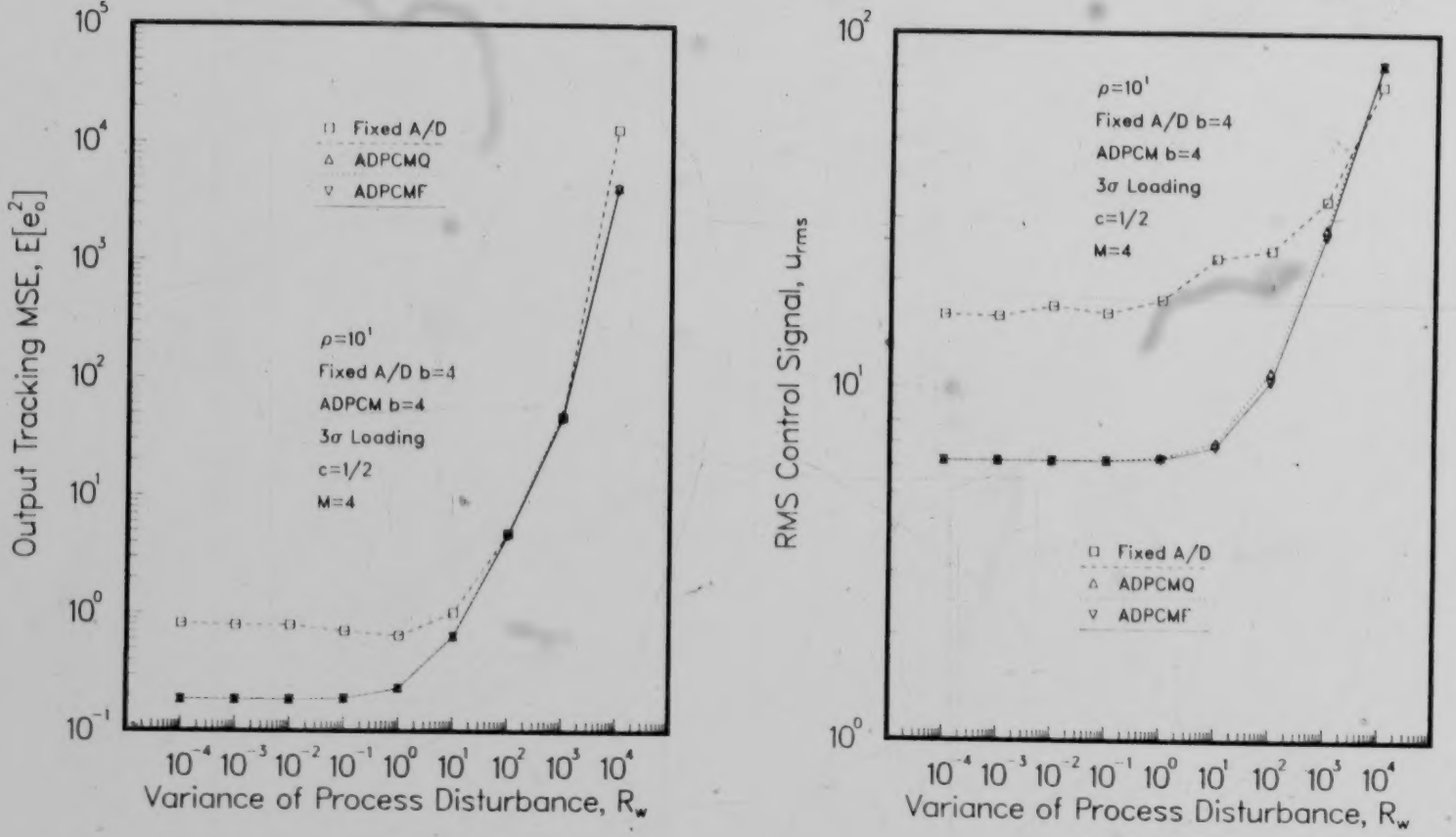


Fig. 5.31 Control System Performance Comparison for Fixed A/D, ADPCMQ, and ADPCMF Coders with $\rho=10$ and Random Set-Points.

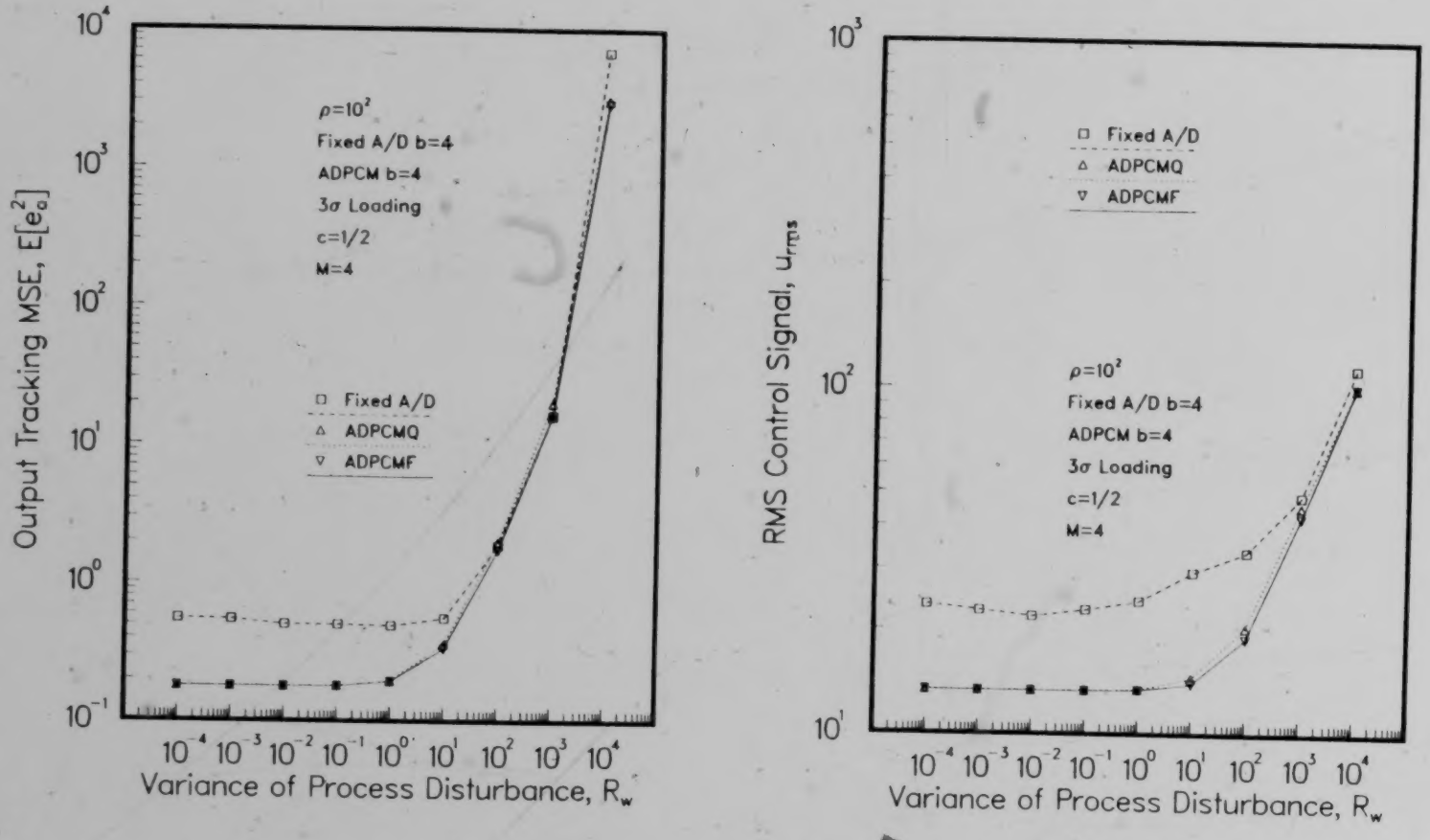


Fig. 5.32 Control System Performance Comparison for Fixed A/D, ADPCMQ, and ADPCMF Coders with $\rho=10^2$ and Random Set-Points.

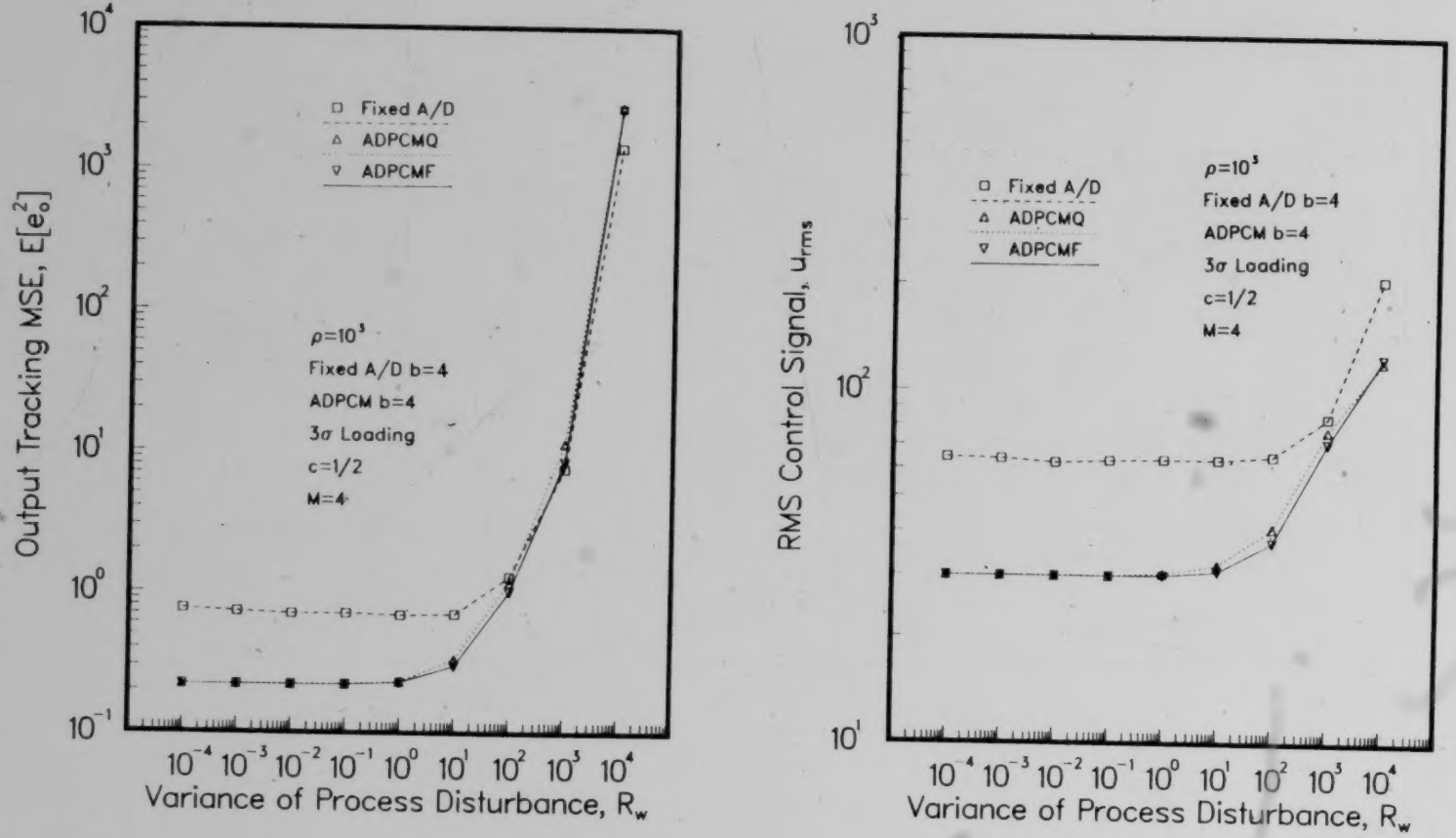


Fig. 5.33 Control System Performance Comparison for Fixed A/D, ADPCMQ, and ADPCMF Coders with $\rho=10^3$ and Random Set-Points.

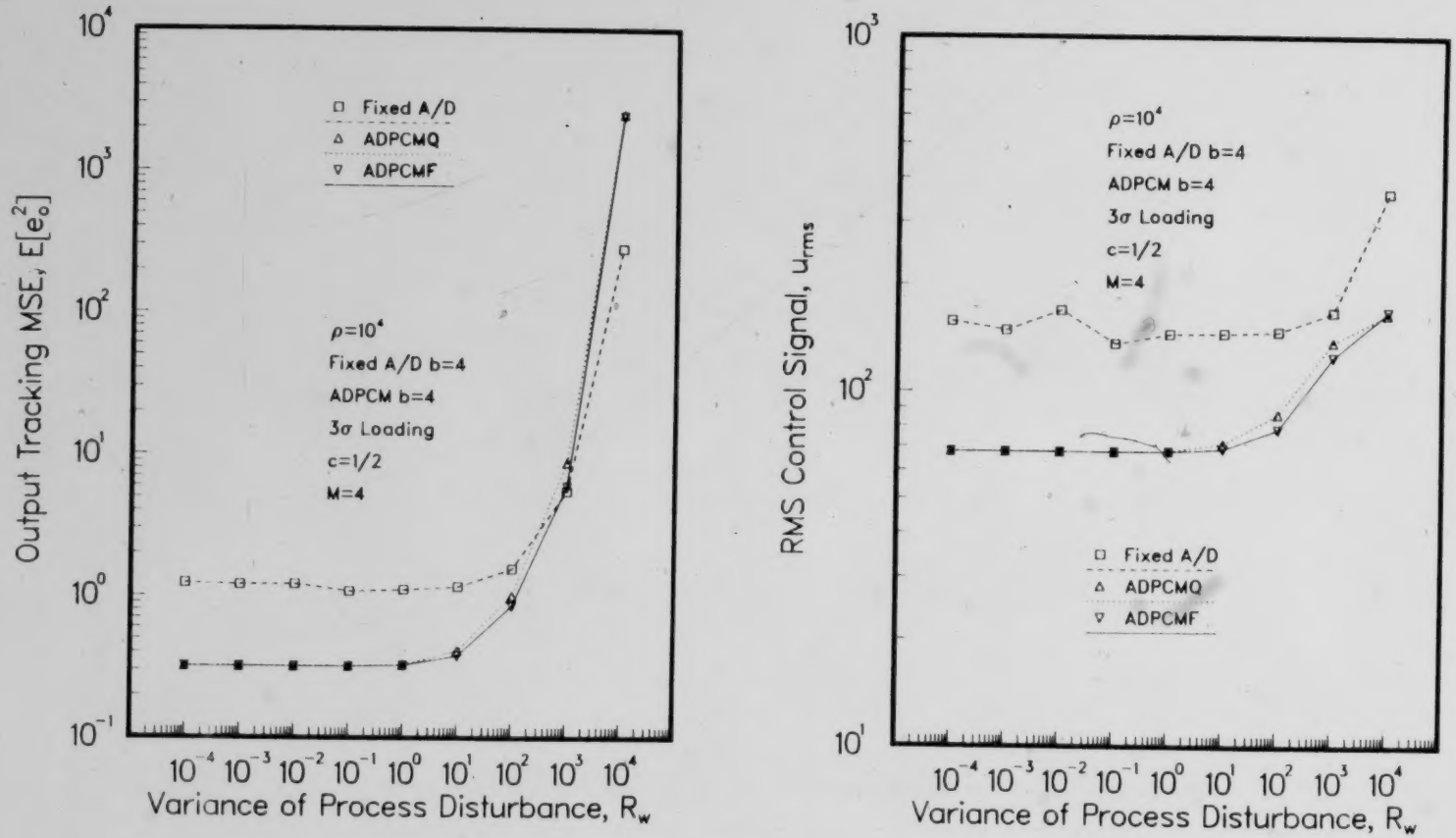


Fig. 5.34 Control System Performance Comparison for Fixed A/D, ADPCMQ, and ADPCMF Coders with $\rho=10^4$ and Random Set-Points.

saving in the RMS control efforts with ADPCM coders compared with that required by the fixed A/D system. In all cases, for low R_w levels (from 10^{-4} to about 1), the mean square tracking error is rather insensitive to the increasing R_w values. This indicates the tracking error is limited by the transient effects of the control system when R_w is low. Starting from $R_w=10$, the tracking error increases significantly as R_w is increasing. This illustrates that the effect of process disturbance at these R_w values becomes more significant to the control system and hence affects the control performance. The RMS control signal behaves similarly for low R_w and also increases when it becomes large. The fixed A/D performance matches to that of the ADPCM coders only at $R_w=10^3$, which is the assumed maximum value used to scale the fixed quantizer. For other R_w, the fixed A/D performance is inferior to that of the ADPCM coders except at $R_w=10^4$ for $\rho=10^3$ and 10^4. At this LQG weighting, the quantizer uses a very conservative scaling factor and hence it results in better performance.

For this random step input response the difference between the performance of ADPCMQ and ADPCMF is almost unnoticeable. ADPCMF, however, still performs better for high R_w and high ρ factors, although the difference is minute. In general, the robustness to R_w is a desirable feature provided by the ADPCM coders. They can reduce not only the output tracking error but also the expenditure of the control efforts. The resulting total cost for the LQG

control system with ADPCM coders will hence be reduced and will be less than that of the system with a fixed A/D converter.

The simulation results in this section are also summarized in Figs. 5.35, 5.36 and 5.37 for the fixed A/D converter, the ADPCMQ coder, and the ADPCMF coder, respectively, for various ρ factors. These figures illustrate again that with higher ρ weightings, a higher control effort would be necessary. However, the output tracking MSE does not necessarily decrease when ρ is increasing for low R_W levels from 10^{-4} to about 1. From the diagrams, it can be observed that for this low range of R_W, the MSE decreases as ρ increases from 1 to 10^2 but then increases when ρ increases from 10^3 to 10^4. This is a result of the highly oscillatory transient responses for the high ρ systems. When R_W is above 1, the tracking error then decreases for increasing ρ. Figs. 5.35 to 5.37 suggest that $\rho=10^2$ will provide the best compromise between the transient behaviour and the time required to reach the steady-state.

5.3.3 ADPCM in an Unstable Plant

The unstable second-order plant (Plant B) used in this simulation has a transfer function (including zero-order-hold) given similarly by (5.15) with $a_1=0.001008361$, $a_2=0.001008361$, $b_1=2.1008361$, and $b_2=-1.000$. This will result in open-loop plant pole locations at $z_1=1.371942$, which is an unstable pole, and $z_2=0.7288941$.

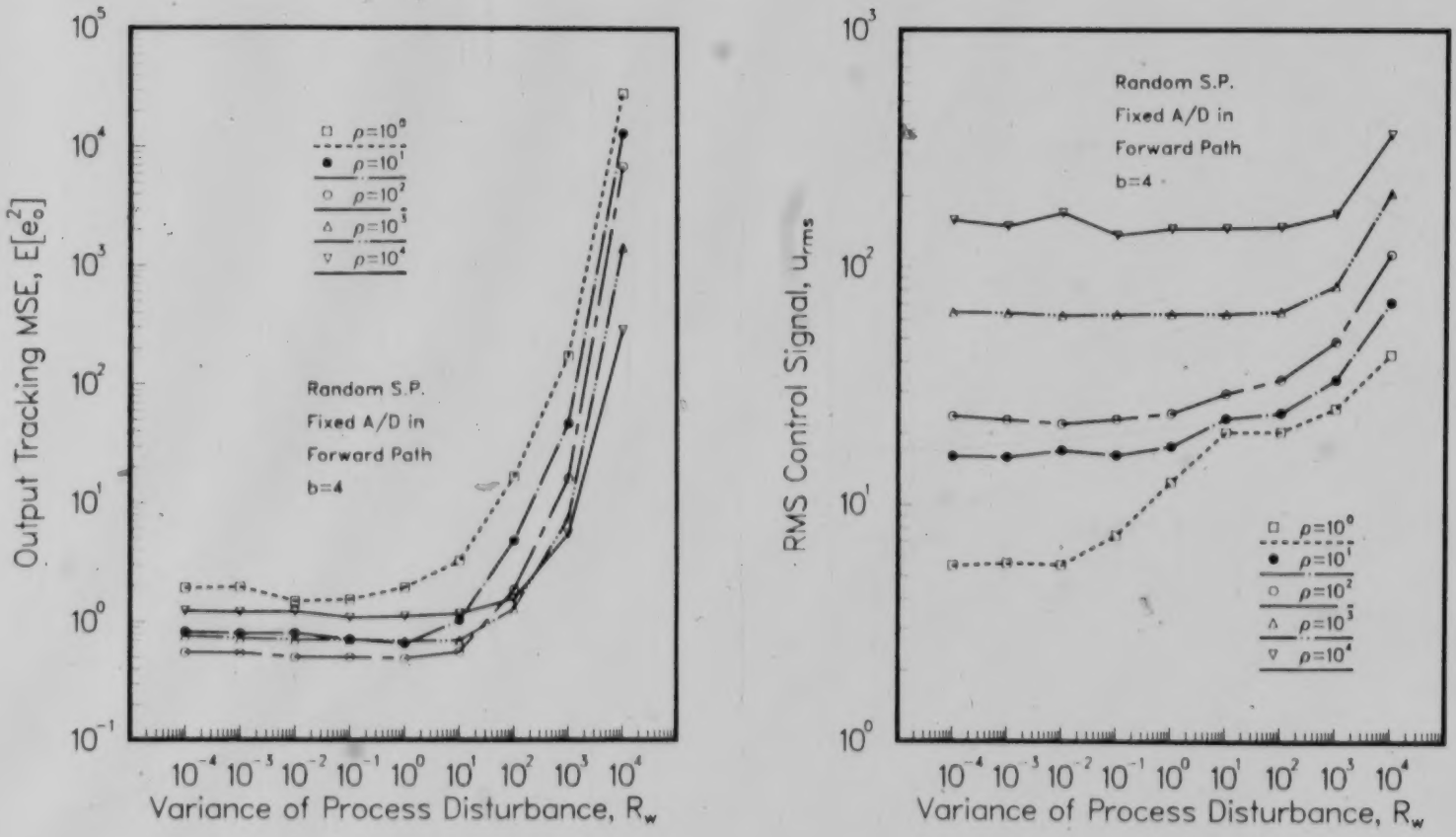


Fig. 5.35 Control System Performance of Fixed A/D for b=4 and Various LQG Weighting Factors ρ with Random Set-Points.

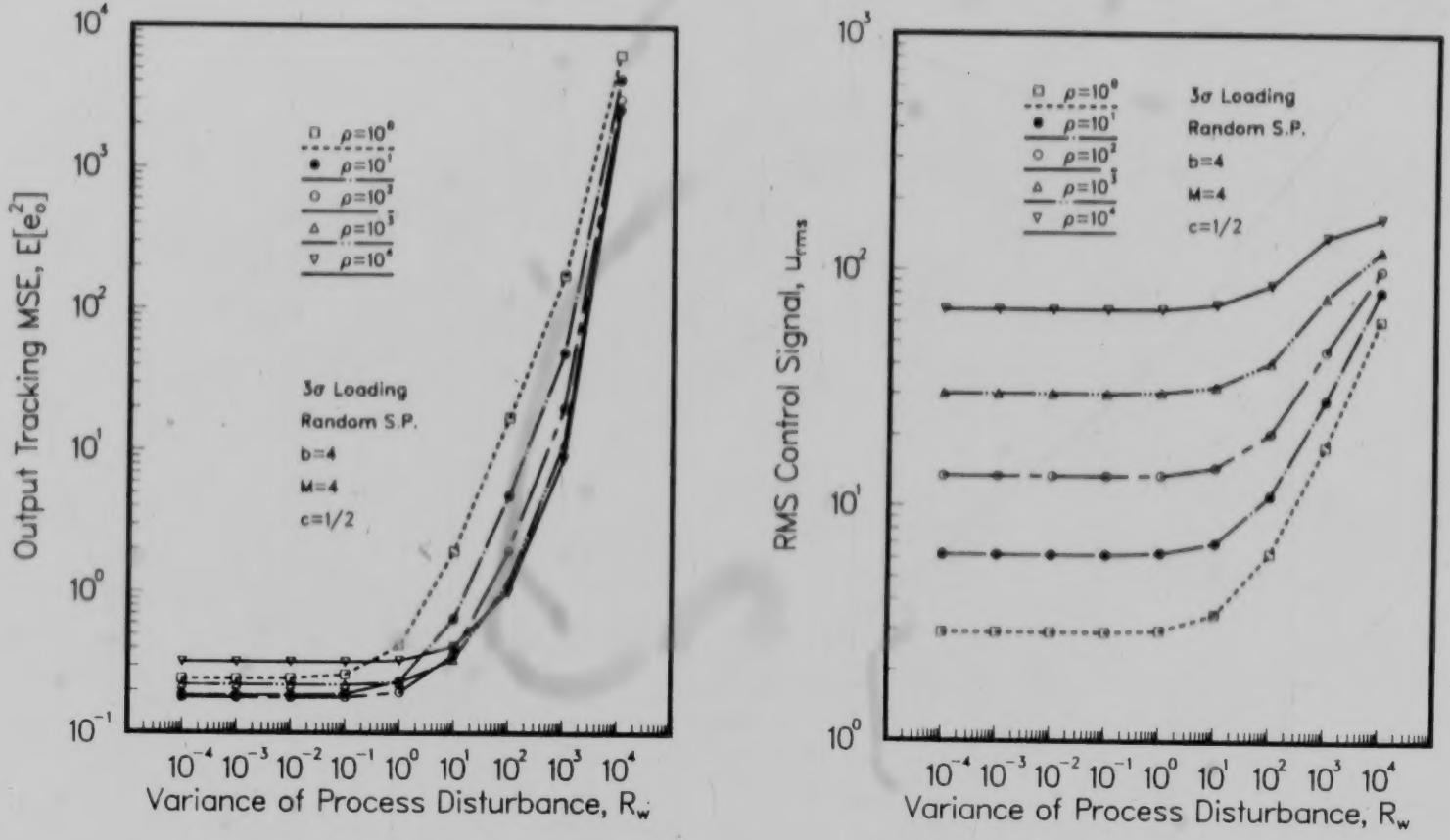


Fig. 5.36 Control System Performance of ADPCMQ Coder for b=4 and Various LQG Weighting Factors ρ with Random Set-Points.

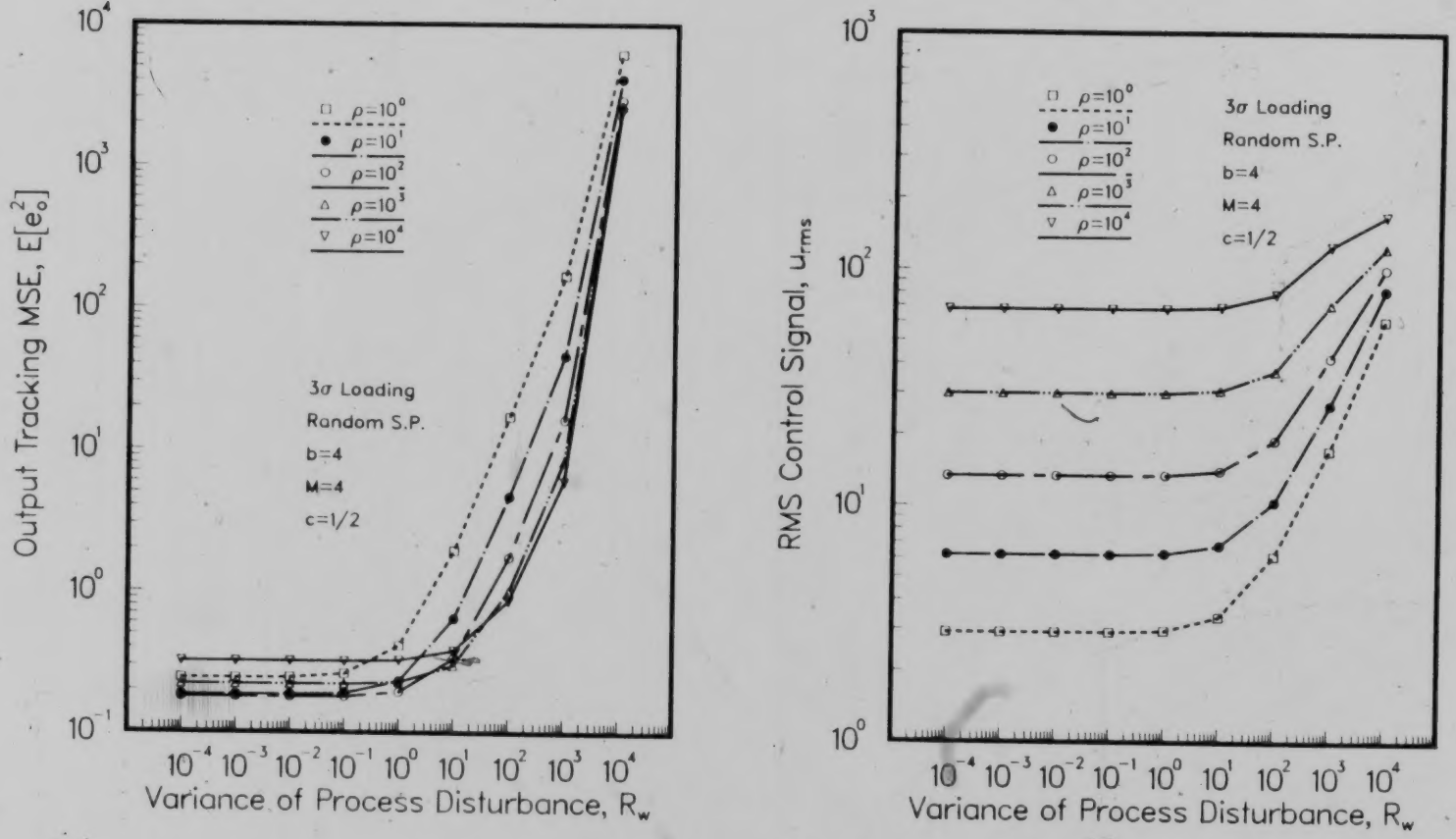


Fig. 5.37 Control System Performance of ADPCMF Coder for b=4 and Various LQG Weighting Factors ρ with Random Set-Points.

For this Type-0 system, a non-zero steady-state error to step input will result. Thus, to reduce this to a zero steady-state error, an offset compensation control term must also be included. This is calculated from (A.5) or (B.1) and can be shown to be $u_0=r(1-b_1-b_2)/(a_1+a_2)$. Realizing the transfer function in a direct observer canonical form, as before, and formalizing it into a state-space equation similar to (5.16), (5.17), and (5.18), the steady-state Kalman filter gain L and the optimal state feedback control gain K can be calculated quite readily.

The solution variance ratio R_s is solved similarly by graphical means as before and is shown in Fig. 5.38. With R_s specified for b=4 and 3σ loading, the LQG compensator will then be completely specified once K and L are computed. Fig. 5.39 shows the simulated closed-loop unit step responses of the unstable system for various LQG weightings ρ. From Fig. 5.39, the transient response for $\rho=10^4$ seems to give the best compromise between the overshoot and the time to reach steady-state. Hence, the following simulations will be limited to this LQG factor only. The maximum allowable R_w value is assumed to be $R_{w_{max}}=10^5$.

5.3.3.1 Performance Versus c

Fig. 5.40 illustrates the simulation results of Plant B to a unit step input for various C values with $\rho=10^4$ and a b=4, M=4, and 3σ loaded ADPCMQ coder. This diagram also shows that the selection of C can be crucial to the performance of the control system. Particularly, when C is

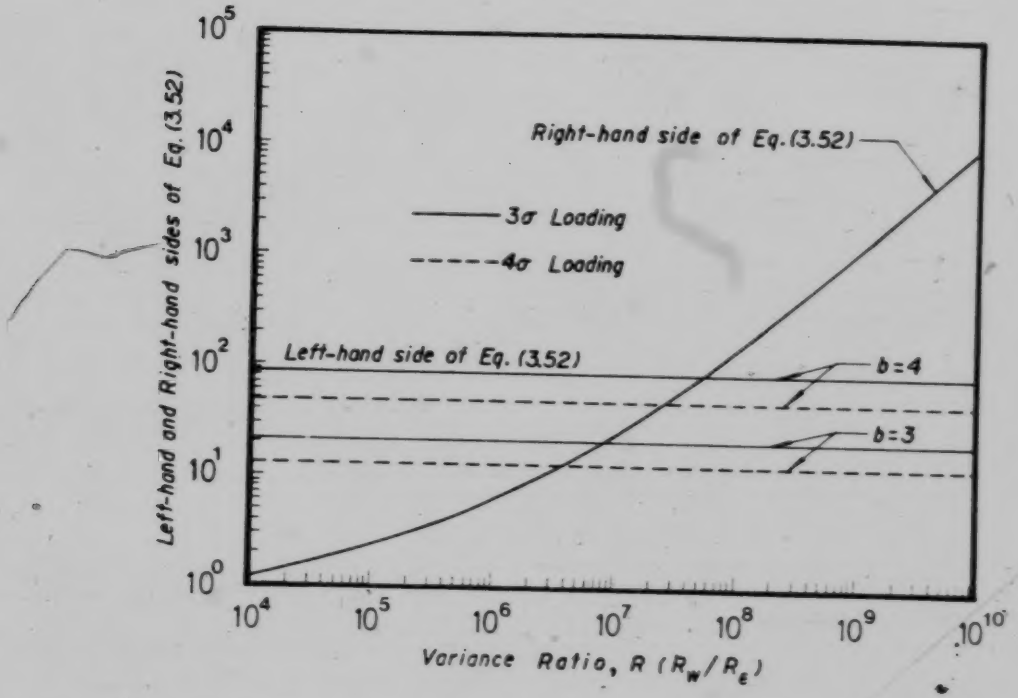


Fig. 5.38 Graphical Solution of Eq.(3.52) for Plant B.

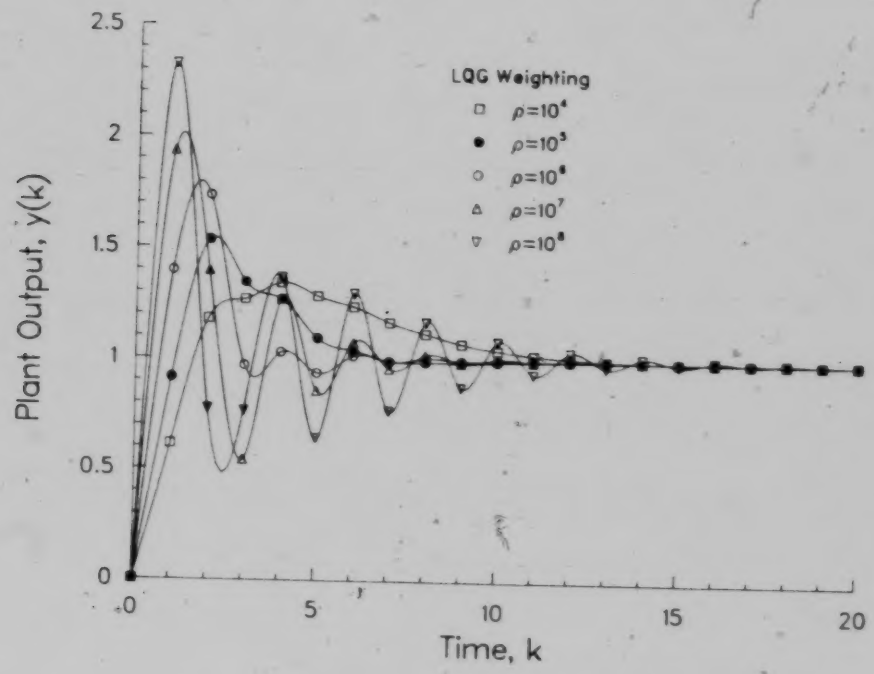


Fig. 5.39 Closed-Loop Unit Step Responses of Plant B for Various LQG Weighting Factors ρ.

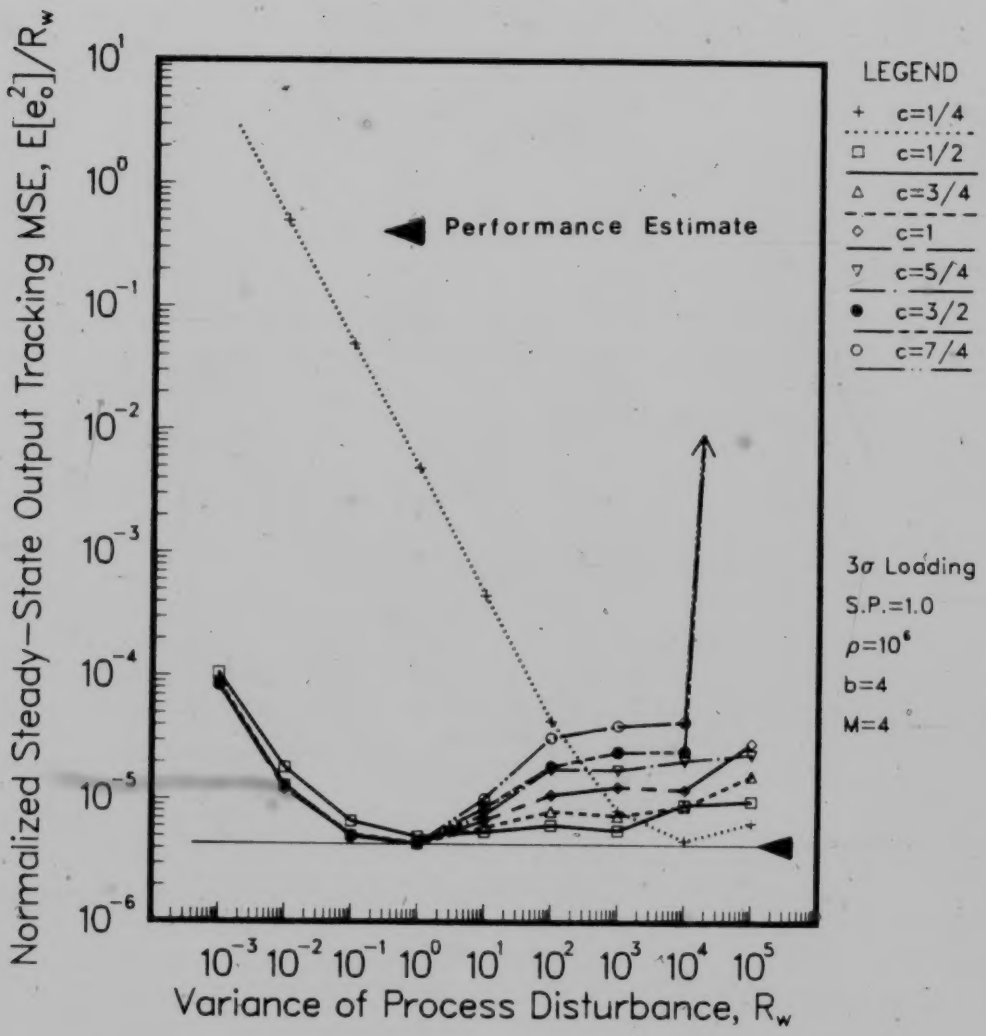


Fig. 5.40 Steady-State Mean-Square-Error of Plant B for Various c Values.

chosen to be 3/2 and 7/4, instability (as indicated by ↑) occurs for $R_W=R_{W_{max}}=10^5$. This illustrates when C is too large, C_T will be too high, resulting in a step-size which is smaller than the optimum value. Consequently, the quantizer is driven into saturation and slope overload very frequently. For this open-loop unstable plant, the effects of these unbounded errors on the control system result in instability as demonstrated in Fig. 5.40. As can be observed from this figure, C should be chosen to be $C=1/2$, as discussed before, to obtain the best compromise between robustness to R_W and agreement to the performance estimate. When $C=1/2$, the steady-state MSE agrees reasonably well with the performance estimate except for high R_W levels (10^4 and 10^5). This is the consequence of the effects of quantizer saturation and slope overload, which occur frequently for high R_W levels, on the control system. This result illustrates the importance of a proper choice of the adaptation threshold parameter in the step-size adaptation algorithm. In the following simulations, C will be chosen from the limiting value of (3.39).

5.3.3.2 Steady-State Performance Comparisons

Fig. 5.41 illustrates the comparisons of steady-state tracking error for Plant B with a fixed A/D converter, an ADPCMQ coder, and an ADPCMF coder in the control loop. The fixed A/D is scaled by the procedure as discussed before. With $\rho=10^4$ and $R_{W_{max}}=10^5$, V_{FS} is selected to be $V_{FS}=r_{max}$ and hence the quantizer is scaled very conservatively. The

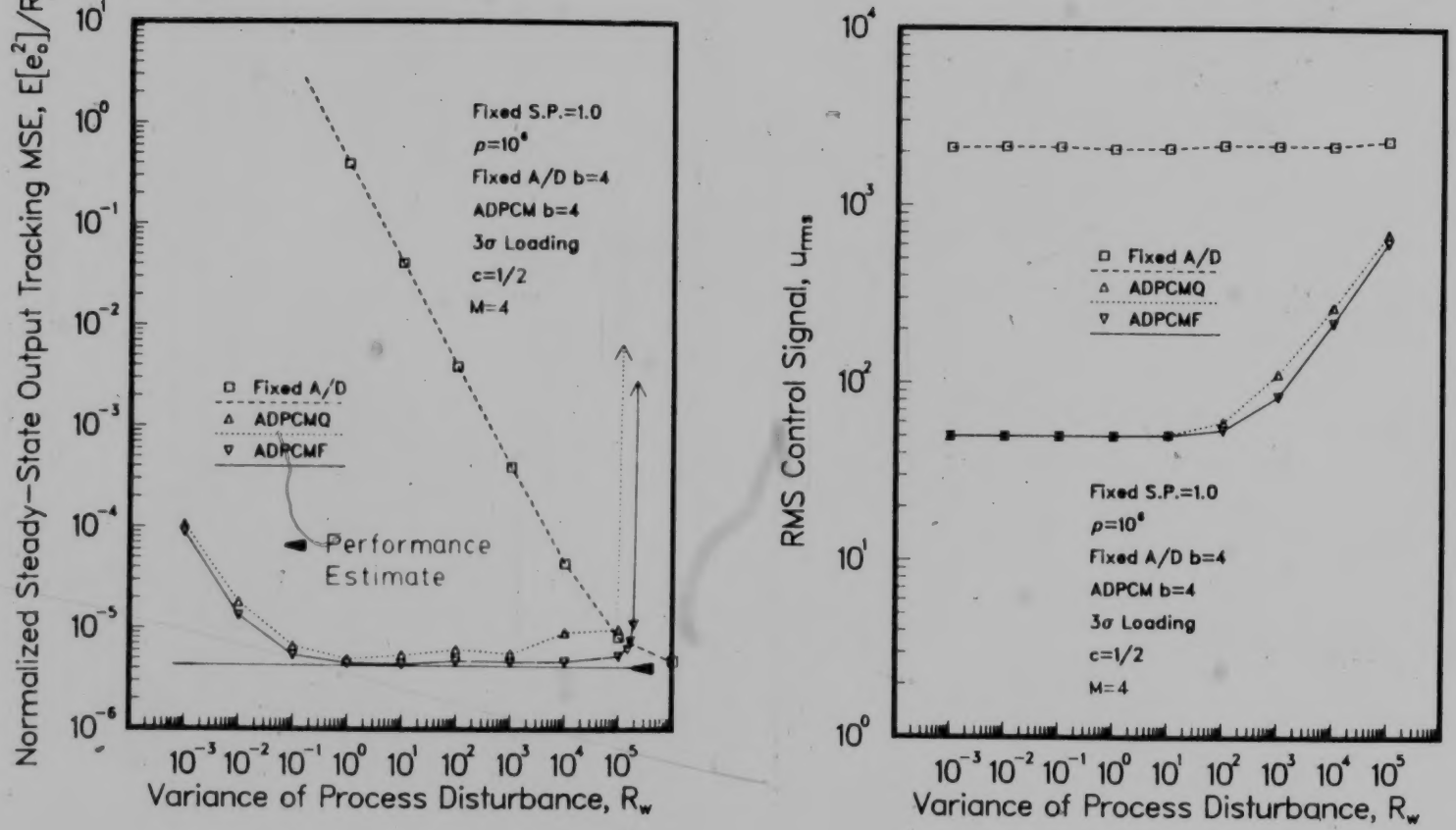


Fig. 5.41 Comparison of Steady-State Mean-Square-Error of Plant B for Fixed A/D, ADPCMQ, and ADPCMF.

result in Fig. 5.41 indicates the robustness of the ADPCM coders to the R_w levels. It also demonstrates that the fixed A/D is very ineffective when applied to a wide range of R_w values. The control signal is rather insensitive to the increase in R_w. Furthermore, for the ADPCM coders, the control signal is also greatly reduced to a much lower value than that of the fixed A/D converter. This also shows that ADPCM coders can reduce the total cost of the LQG control system.

The ADPCMF coder performs better than the ADPCMQ coder and can yield a performance which agrees quite well with the performance estimate for the range of R_w limited by R_s (from 10^5 down to about 1). The control signal is also lower than that of ADPCMQ for $R_w>10$. Even at high R_w levels ($R_w=10^4$ and 10^5), the performance is improved from that of the ADPCMQ coder at the same R_w values, showing the advantage of the use of feedforward information for the step-size adaptation. If better control system performance is desirable, ADPCMF is highly recommended.

The systems with ADPCMF and ADPCMQ coders experienced, however, an instability when $R_w>R_{w_{max}}$ as indicated in Fig. 5.41 by ↑. For ADPCMQ, instability occurred when $R_w=1.2\times10^5$ at about $k=2060$. At that time the system experienced a large disturbance, as shown in Fig. 5.42a, which resulted from the errors due to quantizer slope overload. Instability also occurred for the ADPCMF system, but at a higher process disturbance level. With $R_w=2.5\times10^5$,

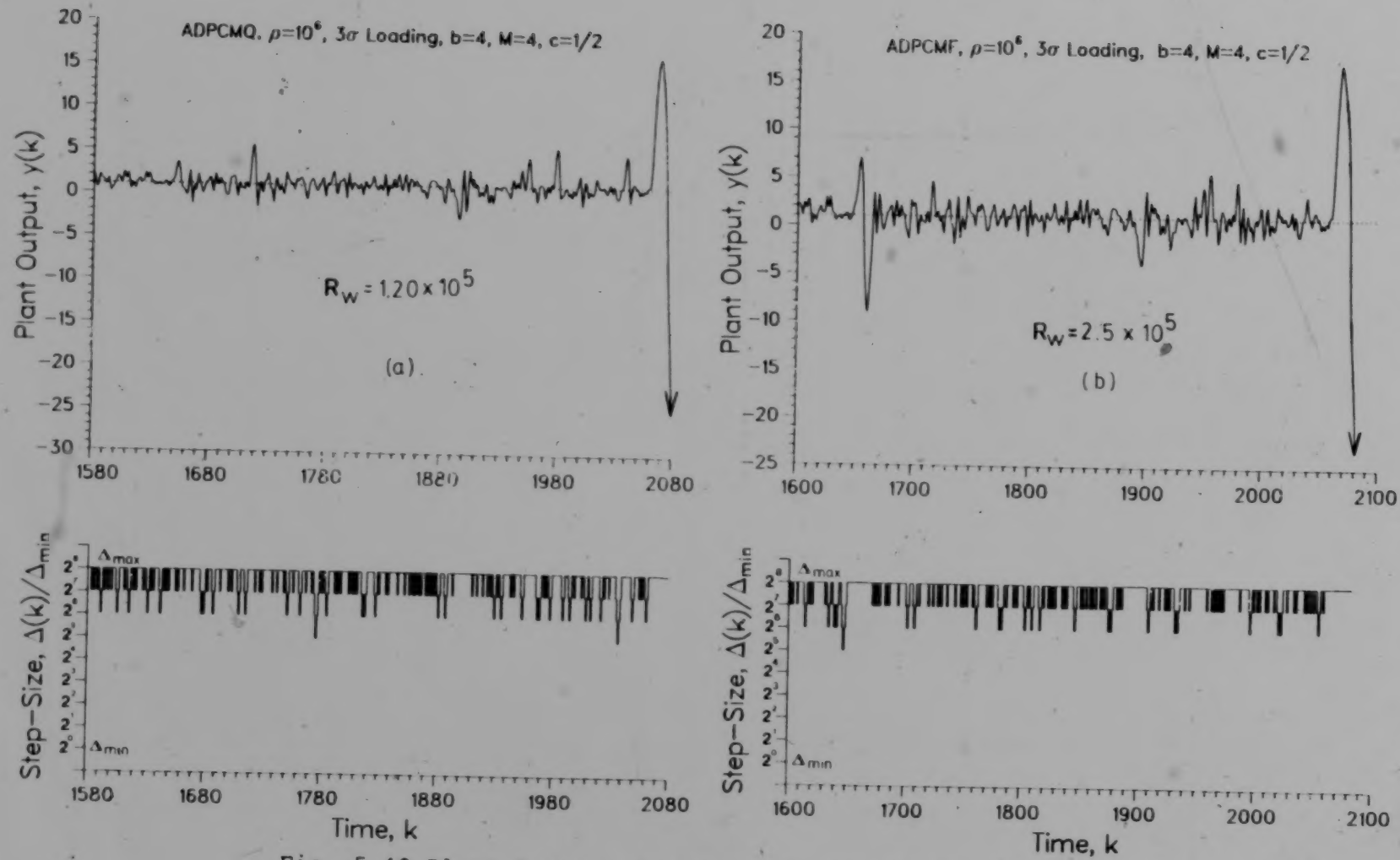


Fig. 5.42 Plant Output and Step-Size Responses of Plant B to a Very Large Process Disturbance.
(a)ADPCMQ, $R_w=1.20\times10^5$. (b)ADPCMF, $R_w=2.5\times10^5$

instability occurred at about K=2055 as a result of slope overload. During the instability, the plant output decreased in the negative direction without bound as shown in Fig. 5.42 indicated by ↓. This instability result demonstrates what can happen if the system is inadequately designed. When designing a system for an open-loop unstable plant, one must be careful to choose the quantizer scaling factor such that the chances of slope overload and quantizer saturation will be very rare.

5.3.3.3 Transient Performance Comparisons

This section presents the final result of the computer simulations. Fig. 5.43 illustrates the mean square tracking error, computed from (5.13), of Plant B to random step inputs which are uniformly distributed between $(0, r_{max})$ with r_{max}=10 as before.

The result demonstrates performance behaviour similar to that in Plant A. For low R_w levels (from 10^{-3} to 10^{3}), the output tracking MSE is insensitive to the increasing R_w values and the performance is essentially limited by the transient errors of the plant output. For higher R_w levels, however, the process disturbance has a more significant effect on the control system and hence a higher MSE results. For $R_w=R_{w_{max}}$ the MSE increase more rapidly as a result of frequent occurrences of quantizer saturation and slope overload which introduce unbounded errors.

It can also be seen from Fig. 5.43 that the robustness of the ADPCM coders can be very effective in reducing the

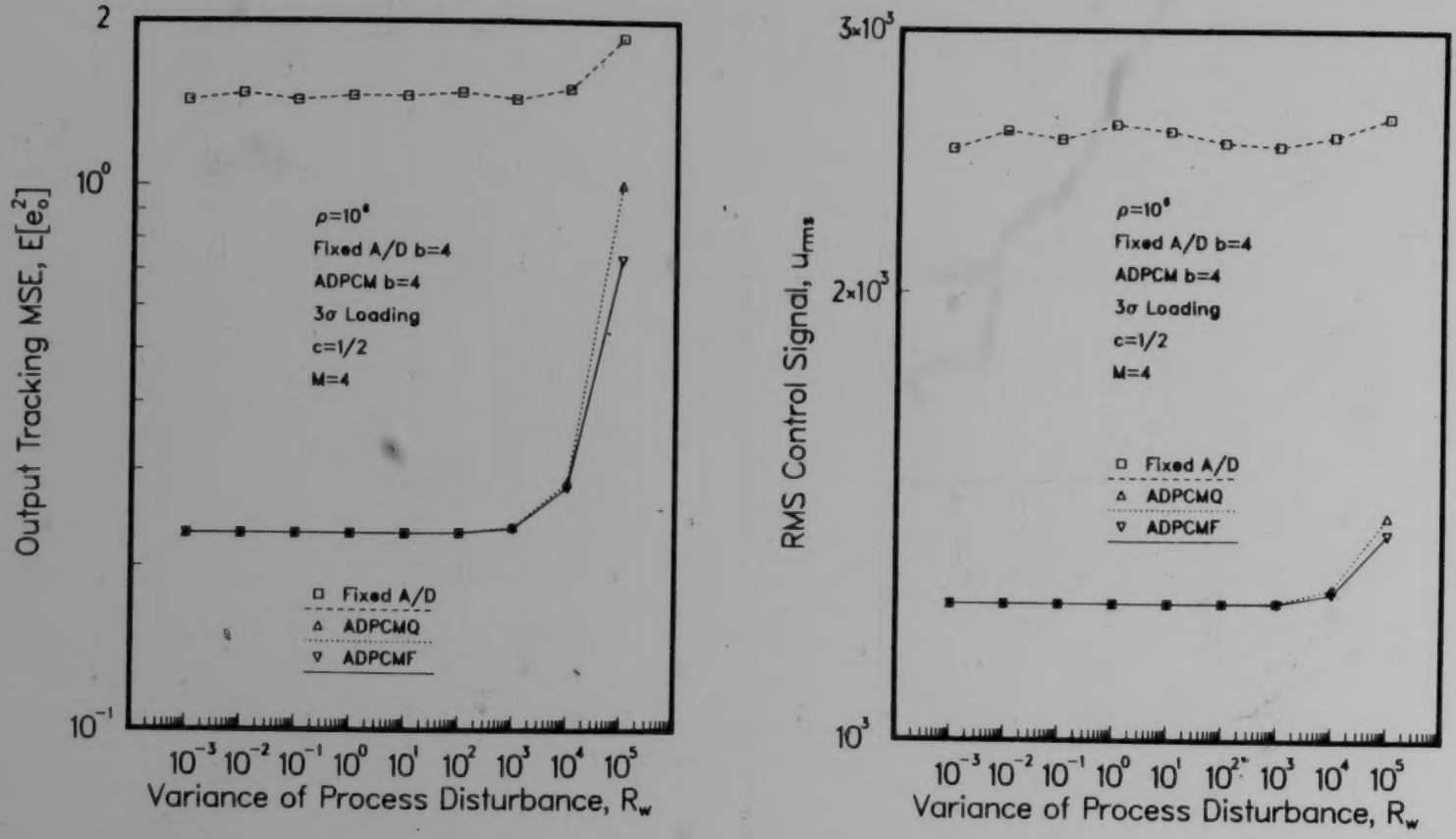


Fig. 5.43 Comparison of Mean-Square-Error of Plant B for Fixed A/D, ADPCMQ, and ADPCMF with Random Set-Points.

tracking error as well as the control signal. For low R_w, the saving in control effort is tremendous (about 100%) and the MSE is also greatly reduced. As a result, ADPCM coders can yield a total cost much lower than that of the fixed A/D converters.

5.4 Summary

In this chapter, an approximate steady-state performance analysis has been carried out and a steady-state performance index in terms of the output tracking mean-square-error has also been obtained. The stability issue of an ADPCM coder in a control loop has also been discussed briefly. It is concluded that there is always a possibility of instability when such a coder is applied to the control of an open-loop unstable plant, as a result of the large unbounded error introduced by quantizer saturation and slope overload. Digital computer simulation results are presented for unit step as well as random step reference inputs and the resulting performances are compared with the values obtained from the theoretical analysis. The results have indicated the superiority of an ADPCM coder when compared with a fixed A/D having the same number of quantizer bits. An ADPCM coder is very effective in reducing the output tracking error and the control signals. Thus the total cost of the LQG system can also be reduced. Limit cycle effects have been observed in the simulations for ADPCMQ coders under certain conditions. This demonstrates

the highly nonlinear effects of the adaptive quantizer in the control loop. Under the same conditions, an ADPCMF coder can avert from the limit cycle effects which occur in the ADPCMQ system. This demonstrates the advantage of the ADPCMF coder which uses the most current information to update its step-sizes. Thus ADPCMF coders may be preferrable in many applications even if it may require more complicated hardware components. The simulation results have demonstrated the occurrence of instability, for the open-loop unstable plant, when the quantizer is driven out of its designed range. This illustrates the detrimental effects that can be caused by inadequate feedback information. As a conclusion, it is found that ADPCM coders, ADPCMF coders in particular, provide the best performance in terms of robustness to R_w, general agreement to the performance estimate obtained from the simplified theoretical analysis, and its ability to reduce the RMS control effort.

Chapter 6

CONCLUSIONS

6.1 Summary and Conclusions

The purpose of this thesis is to study the benefits that can result from an ADPCM coder, when used as an A/D converter, in a digital feedback control system in terms of dynamic range extension and quantization noise reduction when A/D wordlength is limited.

After a brief review of differential encoding and ADPCM coders in waveform communication, the configuration of a control system having an ADPCM coder in the feedback loop was then presented. An LQG approach of compensator design was selected for this control system because of its many desirable features. A steady-state Kalman filter was chosen to be the ADPCM predictor becuase it is capable of yielding optimum state estimates. Then, by assuming the quantizer noise as a white, zero-mean, uncorrelated additive noise, the controller and the predictor transfer functions were derived. A one-word memory step-size adaptation algorithm for the ADPCM coder was presented and the approach to quantizer scaling was also described. Since the control system and the ADPCM coder would eventually be applied to real physical systems, how they could be realized with hardware components should also be considered. Thus suggestions for hardware implementations were also presented. The performance of the control system was

analysed approximately by a linear transfer function approach. In the analysis, the quantization error was assumed to be an uncorrelated, additive, stationary noise with its intersample correlation ignored. Furthermore, quantizer saturation and slope overload effects were not included in the analysis. The resulting performance analysis is simply that of a control system with a nonadaptive DPCM A/D converter which has been scaled properly for each given process disturbance variance. This performance estimate then serves as a lower limit for the performance of ADPCM coders in control systems. Computer simulation results were then obtained and displayed in graphical form for evaluating the merits of the ADPCM system.

The simulation results have indicated that ADPCM quantizers can provide superior performance to that of fixed A/D converters having the same quantizer wordlength. The performance of the control system with ADPCM agrees reasonably well with the performance estimate which has ignored the nonlinear effects of the ADPCM coder. The performance is also found to be robust to a wide range of process disturbance variances. With quantization error reduction provided by the ADPCM coder, the control system output tracking error is also reduced. Furthermore, with an ADPCM coder in the control loop, a significant reduction in the RMS control effort is also achieved. Thus, in general, the total LQG cost of the control system with an ADPCM coder is much lower than that of a control system with a fixed

A/D. Hence with an ADPCM coder, the control system performance can generally be improved. Simulation results have also shown that a forward (ADPCMF) coder always outperforms a feedback (ADPCMQ) coder, although the performance is only slighty improved. In certain situations, however, ADPCMF coders are more desirable. When the variance of the process disturbance is near the assumed maximum value, ADPCMF generally produces better performance than that of ADPCMQ. In the simulations, with an ADPCMQ coder, it was observed that limit cycles existed when a large jump in the process disturbance variance occurred. But, with an ADPCMF coder, limit cycles did not occur for the same process disturbance. The existence of limit cycles demonstrates one of the many potential nonlinear effects on a feedback control system as a result of the ADPCM A/D converter. Closed-loop instability of the control system did not occur for the marginally stable plant when there was inadequate feedback information due to quantizer saturation and slope overload. Instability did occur, however, for the open-loop unstable plant when the process disturbance level was out of the range that could be handled by the quantizer. This demonstrates the necessity of adequate system design, especially for unstable systems.

In general, when an ADPCM coder is applied to a feedback control system, one can expect a significant improvement in the control system performance when compared with that of a system with a fixed A/D converter. ADPCM is a

very useful technique for quantization noise reduction and dynamic range extension. This is especially true when A/D wordlength is limited, such as in cases where fast A/D converters which usually have very short wordlength are employed. As can be observed in the simulations, even with short A/D wordlengths (b=4), the ADPCM performance improvement is significant when compared with that of a fixed A/D converter with the same wordlength. Hence, if longer wordlength is available, a much more significant performance improvement can be expected.

ADPCM is also a very useful approach to realizing a special form of floating-point A/D converter (Oppenheim, 1978) of which the output may be applied directly to fast floating-point processors (Ware *et al.*, 1984) which are useful for implementing the control system. The only limitations are the hardware constraints of the components to be used for implementing the control system. For example, the internal finite wordlength round-off effects may increase the control system error. The accuracy of the ADPCM coder is limited by the accuarcy of the D/A converter used in converting the one-step-ahead plant prediction. For an ADPCMF coder, the converter performance may also depend on the accuracy limit of the analog comparator used in the adaptation threshold decision.

6.2 Suggestions for Further Study

The study reported here has demonstrated that ADPCM A/D conversion is one attractive way of reducing quantization noise effects in feedback control systems, especially when A/D wordlength is rather limited. It has also demonstrated that ADPCM is very useful in extending the useful range of an A/D converter and hence is very useful in applications where one may expect a very noisy environment with disturbances having wide variance variations. This study is, however, only a very modest beginning of an area, viz., the search of efficient ways of improving quantization in feedback control systems, which remains to be explored and investigated.

The design of the ADPCM coder, the control system, and the theoretical performance are much simplified by assuming additive quantization noise so that linear transfer function analysis is applicable. This approach has ignored the intersample correlation, which results from the step-size adaptation, present in the quantization error. The correlation of the process disturbance w and the prediction error δ to the quantization error has also been excluded in the analysis. As a result, the linear transfer function approach can only yield suboptimal system performance. The approximate performance estimate (5.10) is very crude and is only useful for approximating nonadaptive systems. Moreover, the approximate analysis is not able to describe the dynamic behaviour of the system and is not able to indicate the

nonlinear effects, such as limit cycles, which result from the presence of an adaptive quantizer in the feedback loop.

To obtain a more precise description of the dynamic behaviour of the control system and a better performance analysis, more rigorous theoretical analysis is required. This may require a more precise statistical model rather than an additive uniform distribution assumption for the quantization error. Since the quantization noise is a correlated process, joint distribution between δ, w, and ϵ may also be required. Furthermore, in view of the coupling between the input and output of the plant through feedback of the control system, it can be conjectured that the plant system model may also affect the behaviour of the quantization error.

Besides obtaining a better statistical model for the quantization noise, a more precise model for the prediction error δ is also needed since it will be useful in the adaptive quantizer design. By the linear analysis, δ is shown to be independent of the set-point. But simulation results have indicated that under certain conditions, the reference input may affect the behaviour of δ and thus the quantizer behaviour. It may be necessary to investigate the role of the set-point in the dynamic behaviour of δ. With better information about the prediction error δ, it is possible to design a quantizer which can reduce the probability of quantizer saturation and slope overload.

Most important of all, with the adaptive quantizer in the control loop, instability may arise even if the closed-loop system is designed properly such that it would be stable. Thus a more rigorous stability analysis is needed so that guidelines may be available for the design of a stable closed-loop system even when an adaptive quantizer is present in the feedback loop.

Although theoretical analysis results are available in the communication area (Goldstein and Liu, 1977a, b, c), they are not directly applicable to the present control problem. Unlike a control system, a communication system usually does not involve feedback information. Thus, in communication, the source which is to be quantized is not affected by the quantization error. In control systems, however, quantization error is coupled back to the plant via feedback and hence will also appear at the plant output which is the source to be quantized. Therefore, theoretical analysis pertinent to feedback control system is necessary, although some ideas may be borrowed from the communication system area (Goldstein and Liu, 1977a, b, c).

The one-word memory step-size adaptation algorithm used in this research is basically an idea borrowed from communication system which does not have the feedback coupling effect. Hence, it is basically an open-loop design. For a control system, however, due to the feedback coupling effect through the plant, the statistics of the source (plant output) to be quantized also contains the

quantization error introduced by the feedback. Thus the problem becomes a closed-loop design problem and its nature is fairly different from that of a communication system. The one-word memory algorithm may not be ideal for a feedback control system application. Therefore, step-size algorithms suitable for feedback control systems other than the one-word memory algorithm may need be explored; perhaps, by including more memory words in the algorithm, or by matching the algorithm to the statistics of the quantizer input, which is the prediction error δ.

Finally, this study is motivated by problems arising from hardware implementations of digital control systems. Thus it would be very useful to actually test the ADPCM A/D converter in a control system with real practical hardware components. However, since all hardware components have finite wordlengths, internal round-off errors cannot be ignored. Thus, theoretical analysis of internal finite wordlength effects, a difficult problem in its own right, and their interactions with the adaptive quantization effects should also be considered. The results from this analysis would be very useful as guidelines for selecting the internal register and A/D wordlengths when implementing the system with practical hardware components.

BIBLIOGRAPHY

Adelmann, H. W., Y. C. Ching, and B. Gotz (1979). An ADPCM approach to reduce the Bit Rate of μ-law Encoded Speech. *Bell Syst. Tech. J.*, 58, 1659-1671.

Analog Devices (1982). *Data Acquisition Databook 1982.* Analog Devices, Norwood, Massachusetts.

Anderson, B. D. O. and J. B. Moore (1979). *Optimal Filtering.* Prentice-Hall, Englewood Cliffs, New Jersey.

Åstrom, K. J., E. I. Jury, and R. G. Agniel (1970). A Numerical Method for the Evaluation of Complex Integrals. *IEEE Trans. Automat. Contr.*, AC-15, 468-471.

Atal, B. S. and M. R. Schroeder (1970). Adaptive Predictive Coding of Speech Signals. *Bell Syst. Tech. J.*, 49, 1973-1986.

Bertram, J. E. (1958). The Effect of Quantization in Sampled-Feedback Systems. *Trans. AIEE*, 77, Part II, 177-181.

Boddie, J. R. *et al.* (1981). Adaptive Differential Pulse-Code-Modulation Coding. *Bell Syst. Tech. J.*, 60, 1547-1561.

Brogan, W. (1974). *Modern Control Theory.* Quantum Publishers, New York.

Bryson, A. E. and Y. C. Ho (1969). *Applied Optimal Control.* Blaisdell, Waltham, Massachusetts.

Cohn, D. L. and J. L. Melsa (1975). The Relationship Between an Adaptive Quantizer and a Variance Estimator. *IEEE Trans. Inform. Theory*, IT-21, 669-671.

Cummiskey, P., N. S. Jayant, and J. L. Flanagan (1973). Adaptive Quantization in Differential PCM Coding of Speech. *Bell Syst. Tech. J.*, 52, 1105-1118.

Curry, R. E. (1970). *Estimation and Control with Quantized Measurements.* M.I.T. Press, Cambridge, Massachusetts.

Davisson, L. D. (1972). Rate-Distortion Theory and Application. *Proc. IEEE*, **60**, 800-808.

Dubnowski, J. (1978). A Microprocessor Log PCM/ADPCM Code Converter. *IEEE Trans. Commun.*, **COM-26**, 660-664.

Einarsson, G. (1981). A Robust Adaptive Quantizer with Extended Dynamic Range. *IEEE Trans. Commun.*, **COM-29**, 830-836.

Elias, P. (1955). Predictive Coding — Parts I & II. *IRE Trans. Inform. Theory*, **IT-1**, 16-33.

Fischer, T. R. and D. J. Tinnin (1982). Quantized Control with Differential Pulse Code Modulation. *Proc. IEEE Conf. Decision and Control*, 1222-1227.

Fischer, T. R. and D. J. Tinnin (1984). Quantized Control Using Differential Encoding. *Optimal Control Aapplication & Mehtods*, 5, 69-83.

Flanagan, J. L. *et al.* (1979). Speech Coding. *IEEE Trans. Commun.*, **COM-27**, 710-736.

Franklin, G. F. and J. D. Powell (1980). *Digital Control of Dynamic Systems.* Addison-Wesley, Reading, Massachusetts.

Gersho, A. (1978). Principles of Quantization. *IEEE Trans. Circuit Syst.*, **CAS-25**, 427-436.

Gibson, J. D., S. K. Jones, and J. L. Melsa (1974). Sequentially Adaptive Prediction and Coding of Speech Signals. *IEEE Trans. Commun.*, **COM-22**, 1789-1797.

Gibson, J. D. (1978). Sequentially Adaptive Backward Prediction in ADPCM Coders. *IEEE Trans. Commun.*, **COM-26**, 145-150.

Gibson, J. D. (1980a). Adaptive Prediction in Speech Differential Encoding Systems. *Proc. IEEE*, **68**, 488-525.

Gibson, J. D. (1980b). Quantization Noise Filtering in ADPCM Systems. *IEEE Trans. Syst., Man, Cybern.*, **SMC-10**, 529-536.

Goldstein, L. and B. Liu (1976). An ADPCM Realization of Nonrecursive Digital Filters. *IEEE Trans. Acoust., Speech, Signal Processing*, **ASSP-24**, 312-320.

Goldstein, L. and B. Liu (1977a). Quantization Noise in ADPCM Systems. *IEEE Trans. Commun.*, **COM-25**, 227-238.

Goldstein, L. and B. Liu (1977b). Quantization Error and Step-size Distributions in ADPCM. *IEEE Trans. Inform. Theory*, **IT-23**, 216-223.

Goldstein, L. and B. Liu (1977c). Deterministic and Stochastic Stability of Adaptive Differential Pulse Code Modulation. *IEEE Trans. Inform. Theory*, **IT-23**, 445-453.

Goodman, D. J. and A. Gersho (1974). Theory of an Adaptive Quantizer. *IEEE Trans. Commun.*, **COM-22**, 1037-1045.

Goodman, D. J. and R. M. Wilkinson (1975). A Robust Adaptive Quantizer. *IEEE Trans. Commun.*, **COM-23**, 1362-1365.

Gunn, J. E. and A. P. Sage (1973). Speech Data Rate Reduction — Part I: Applicability of Modern Estimation Theory. *IEEE Trans. Aerosp. Electron. Syst.*, **AES-9**, 130-138.

Isermann, R. (1981). *Digital Control Systems*. Springer-Verlag, New York.

Jayant, N. S. (1970). Adaptive Delta Modulation With a One-Bit Memory. *Bell Syst. Tech. J.*, **49**, 321-343.

Jayant, N. S. (1973). Adaptive Quantization With a One-Word Memory. *Bell Syst. Tech. J.*, **52**, 1119-1144.

Jayant, N. S. (1974). Digital Coding of Speech Waveforms: PCM, DPCM, and DM Quantizers. *Proc. IEEE*, **62**, 611-632.

Jayant, N. S. (Ed.) (1976). *Waveform Quantization and Coding*. IEEE Press, New York.

Johnston, J. D. and K. J. Goodman (1978). Multipurpose Hardware for Digital Coding of Audio Signals. *IEEE Trans. Commun.*, **COM-26**, 1785-1788.

Knowles, J. B. and R. E. Edwards (1965). Effect of a Finite-Word-Length Computer in a Sampled-Data Feedback System. *Proc. IEE*, **112**, 1197-1207.

Kuo, B. C. (1980). *Digital Control Systems*. Holt, Rinehart and Winston, New York.

Kwakernaak, H. and R. Sivan (1972). *Linear Optimal Control Systems*. Wiley-Interscience, New York.

Makhoul, J. (1975). Linear Prediction: A Tutorial Review. *Proc. IEEE*, **62**, 561-580.

Max, J. (1960). Quantizing for Minimum Distortion. *IRE Trans. Inform. Theory*, **IT-6**, 7-12.

Moroney, P. (1983). *Issues in the Implementation of Digital Feedback Compensators*. M.I.T. Press, Cambridge, Massachusetts.

Melsa, J. L. and R. B. Kolstad (1977). Kalman Filtering of Quantization Error in Digitally Processed Speech. *Conf. Rec. 1977 Int. Conf. Communications*, 310-313.

Noll, P. (1974). Adaptive Quantizing in Speech Coding Systems. *Proc. International Zürich Seminar on Digital Commun.*, B3(1)-B3(6).

Noll, P. (1975a). A Comparative Study of Various Quantization Schemes for Speech Encoding. *Bell Syst. Tech. J.*, **54**, 1597-1614.

Noll, P. (1975b). The Performance of PCM and DPCM Speech Coders in the Presence of Independent and Correlated Errors. *Conf. Rec. Int. Conf. on Commun.*, pp. 30-1 – 30-5.

Ogata, K. (1970). *Modern Control Engineering.* Prentice-Hall, Englewood Cliffs, New Jersey.

O'Neal, J. B., Jr. (1966a). Predictive Quantizing Systems (Differential Pulse Code Modulation) for the Transmission of Television Signals. *Bell Syst. Tech. J.*, **45**, 689-721.

O'Neal, J. B., Jr. (1966b). Delta Modulation Quantization Noise Analytical and Computer Simulation Results for Gaussian and Television Input Signals. *Bell Syst. Tech. J.*, **45**, 117-141.

Oppenheim, A. V. (1970). Realization of Digital Filters Using Block-Floating-Point Arithmetic. *IEEE Trans. Audio Electroacoustics*, **AU-18**, 130-136.

Oppenheim, A. V. (Ed.) (1978). *Applications of Digital Signal Processing.* Prentice-Hall, Englewood Cliffs, New Jersey.

Oppenheim, A. V. and R. W. Schafer (1975). *Digital Signal Processing.* Prentice-Hall, Englewood Cliffs, New Jersey.

Paez, M. D. and T. H. Glisson (1972). Minimum Mean-Squared-Error Quantization in Speech PCM and DPCM Systems. *IEEE Trans. Commun.*, **COM-20**, 225-230.

Peled, A. and B. Liu (1976). *Digital Signal Processing: Theory, Design, and Implementation.* Wiley, New York.

Rabiner, L. R. and B. Gold (1975). *Theory and Applicaton of Digital Signal Processing.* Prentice-Hall, Englewood Cliffs, New Jersey.

Rabiner, L. R. and R. W. Schafer (1978). *Digital Processing of Speech Signals*. Prentice-Hall, Englewood Cliffs, New Jersey.

Rink, R. E. and H. Y. Chong (1975). Finite Wordlength Effects in Digital Control. *Proc. 1975 Canadian Conf. on Automatic Control*. (Univ. British Columbia, June 1975).

Rink, R. E. and H. Y. Chong (1979). Performance of State Regulator Systems with Floating-Point Computation. *IEEE Trans. Automat. Contr.*, **AC-24**, 411-421.

Rink, R. E. (1982). Predictive Quantization for Efficient A/D Conversion. *Int. J. Control*, **36**, 1021-1031.

Sheingold, D. H. (Ed.) (1977). *Analog-Digital Conversion Notes*. Analog Devices, Norwood, Massachusetts.

Song, C. L., J. Garodnick, and D. L. Schilling (1971). A Variable Step-Size Robust Delta Modulator. *IEEE Trans. Commun.*, **COM-19**, 1033-1044.

Slaughter, J. B. (1964). Quantization Errors in Digital Control Systems. *IEEE Trans. Automat. Contr.*, **AC-9**, 70-74.

Spang, H. A., III and P. M. Schultheiss (1962). Reduction of Quantizing Noise by Use of Feedback. *IRE Trans. Commun. Syst.*, **CS-10**, 373-380.

Un, C. K. and M. H. Cynn (1980). A Performance Comparison of ADPCM and ADM Coders. *Conf. Rec. Nat'l Telecomm. Conf.*, 50.1.1-50.1.5.

van Wingerden, A. J. M. and W. L. De Konig (1984). The Influence of Finite Word Length on Digital Control. *IEEE Trans. Automat. Contr.*, **AC-29**, 385-391.

Ware, *et al.* (1984). Fast 64-Bit Chip Set Gangs Up for Double-Precision Floating-Point Work. *Electronics*, **47** (14), 99-103.

APPENDIX A - Derivation of Steady-State LQG Compensator Transfer Function for a System with Non-Zero Set-Point

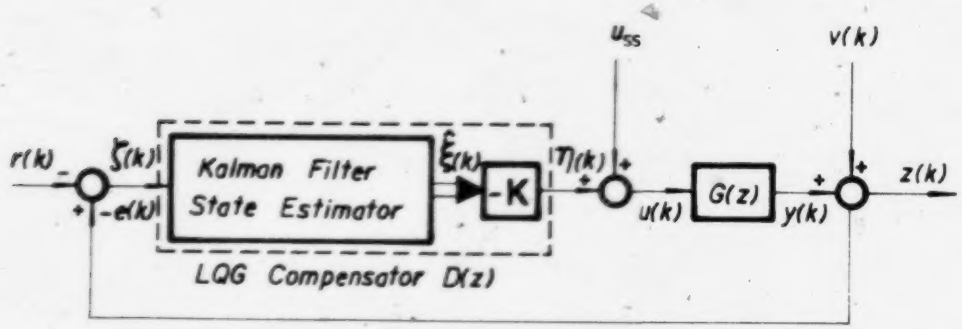


Fig. A.1 LQG Control System with Non-Zero Set-Point [After Moroney (1983)].

Assume a constant set-point† such that

$$r(k) = y_r . \tag{A.1}$$

The output tracking error is given by

$$e(k) = -\zeta(k) = y_r - z(k) \tag{A.2}$$

which is used in the compensator for generating the control sequence $u(k)$. The purpose of the steady-state LQG tracking error regulator problem is to minimize the loss function (3.9). At the steady-state, the following relationship must

---

†For a time-varying reference input $r(k)$, a constant ρ and steady-state time-invariant LQG compensator may not suffice.

hold (from Eqs. (3.1) and (3.2))†

$$\mathbf{x}_{ss} = \Phi\mathbf{x}_{ss} + \Gamma u_{ss} \tag{A.3}$$

$$y_{ss} = \mathbf{H}\mathbf{x}_{ss} \tag{A.4}$$

from which

$$u_{ss} = [\mathbf{H}(\mathbf{I}-\Phi)^{-1}\Gamma]^{-1}y_{ss}. \tag{A.5}$$

The term $[\mathbf{H}(\mathbf{I}-\Phi)^{-1}\Gamma]$ is simply the steady-state gain of the open-loop transfer function. For a Type-1 plant, since there is a pole at $z=1$, $[\mathbf{H}(\mathbf{I}-\Phi)^{-1}\Gamma]$ will become infinite at steady-state and this implies a zero steady-state control signal. For a Type-0 system, u_{ss} is non-zero and its value is given by (A.5). To reduce the tracking error $e(k)$ to zero at steady-state, one must apply this non-zero control signal to the plant so that $y_{ss}=y_r$. Subtracting the steady-state values of (A.3) and (A.4) from (3.1), one would obtain

$$\mathbf{x}(k+1) - \mathbf{x}_{ss} = \Phi[\mathbf{x}(k) - \mathbf{x}_{ss}] + \Gamma u(k) - [\mathbf{x}_{ss} - \Phi\mathbf{x}_{ss}] + \Gamma_1 w(k) \tag{A.6}$$

$$y(k) - y_{ss} = \mathbf{H}\mathbf{x}(k) - y_r. \tag{A.7}$$

In (A.7), it has been assumed that $y_{ss}=y_r$ at the steady-state. Now define new plant states and variables which are the deviations from their steady-state values (Kwakernaak and Sivan, 1972; Moroney, 1983):

---

†The subscripts 'SS' denote constant steady-state values.

$$\xi(k) = \mathbf{x}(k) - \mathbf{x}_s, \tag{A.8}$$

$$\eta(k) = u(k) - u_s, \tag{A.9}$$

$$\gamma(k) = y(k) - y_s$$

$$= y(k) - y_r. \tag{A.10}$$

With these new states (A.6) and (A.7) then become

$$\xi(k+1) = \Phi\xi(k) + \Gamma\eta(k) + \Gamma_1 w(k), \tag{A.11}$$

$$\gamma(k) = H\xi(k), \tag{A.12}$$

$$\zeta(k) = z(k) - y_r$$

$$= \gamma(k) + v(k), \tag{A.13}$$

where $\zeta(k)$ is the new noisy measurement. Eqs. (A.11) through (A.13) have exactly the same form as the original (3.1) and (3.2) except that the plant states are different. Therefore, the design procedure for the LQG compensator is the same as the original regulator problem. The Kalman filter equation for the new states is

$$\hat{\xi}(k|k) = \bar{\xi}(k|k-1) + L[\zeta(k) - H\bar{\xi}(k|k-1)], \tag{A.14}$$

$$\bar{\xi}(k+1|k) = \Phi\hat{\xi}(k|k) + \Gamma\eta(k), \tag{A.15}$$

where L is the steady-state Kalman filter gain vector. The optimal control gain is obtained from

$$\eta(k) = -K\hat{\xi}(k|k), \tag{A.16}$$

where K is the steady-state optimal state feedback control gain vector. Taking the z-transform of Eqs. (A.14) through

(A.16) and solving for $\hat{\xi}(z)$ yields

$$\hat{\xi}(z) = (zI - \Phi + LH\Phi)^{-1}[(I - LH)\Gamma\eta(z) + zL\zeta(z)]. \tag{A.17}$$

From (A.16)

$$\eta(z) = -K\hat{\xi}(z). \tag{A.18}$$

Solving $\eta(z)$ from (A.2), (A.17), and (A.18) one would obtain

$$D(z)=\frac{\eta(z)}{e(z)}=-\frac{\eta(z)}{\zeta(z)}=\frac{zK[zI-\Phi+LH\Phi]^{-1}L}{1+K[zI-\Phi+LH\Phi]^{-1}[I-LH]\Gamma} \tag{A.19}$$

which is the required steady-state LQG compensator transfer function. To generate the actual control signal, one will have to add the extra term u_{ss}, which may be regarded as the steady-state plant output offset compensation term, to the compensator output:

$$u(k) = \eta(k) + u_{ss}. \tag{A.20}$$

For a Type-1 system u_{ss} is zero and is non-zero for a Type-0 system.

APPENDIX B - Derivation of Equations (3.40) and (3.41)

From Appendix A, it is shown that an offset term u_{ss} must be added to the control signal in order to reduce the steady-state plant output tracking error to zero for Type-0 systems. For constant set-points, this offset is also constant and is given by (A.5). In general, one may consider u_{ss} as a scaled reference feedforward signal given by

$$u_{ss}(k) = [\mathbf{H}(\mathbf{I}-\Phi)^{-1}\Gamma]^{-1} r(k) = u_0 r(k), \qquad (B.1)$$

where $r(k)$ is the reference input set-point and $u_0=[\mathbf{H}(\mathbf{I}-\Phi)^{-1}\Gamma]^{-1}$. Thus, with reference to Fig. 3.1 and for constant reference set-points, the control signal, together with the offset compensation term, will be given by

$$u(z) = D(z)e(z) + u_0 r(z). \qquad (B.2)$$

From the block diagram Fig. 3.1 and assuming a deterministic problem, one can write down the z-transform of the prediction error:

$$\begin{aligned} \delta(z) &= y(z) - \bar{y}(z) \\ &= y(z) - F(z)u(z) - P(z)\hat{y}(z). \end{aligned} \qquad (B.3)$$

Substituting (B.2) into (B.3),

$$\delta(z) = y(z) + [F(z)D(z) - P(z)]\hat{y}(z) - F(z)D(z)r(z) - F(z)u_o r(z). \quad (B.4)$$

From Fig. 3.1, the plant output can expressed as

$$y(z) = D(z)G(z)[r(z) - \hat{y}(z)] + G(z)w(z) + G(z)u_o r(z). \quad (B.5)$$

Putting (B.5) into (B.4),

$$\delta(z) = [F(z)D(z) - P(z) - D(z)G(z)]\hat{y}(z) + G(z)w(z) + [D(z)G(z) - F(z)D(z)]r(z) + [G(z) - F(z)]u_o r(z). \quad (B.6)$$

From Eq. (3.23), $F(z)=[1-P(z)]G(z)$, it follows immediately that

$$G(z) - F(z) = P(z)G(z). \quad (B.7)$$

Using (B.7) and (3.23) to simplify $\delta(z)$, (B.6) reduces to

$$\delta(z) = G(z)w(z) + P(z)G(z)u_o r(z) + P(z)D(z)G(z)r(z) - [1 + D(z)G(z)]P(z)\hat{y}(z). \quad (B.8)$$

From the block diagram

$$\hat{y}(z) = \bar{y}(z) + \delta_q(z) = F(z)u(z) + P(z)\hat{y}(z) + \delta_q(z). \quad (B.9)$$

Solving (B.9) for $\hat{y}(z)$, it follows that

$$\hat{y}(z) = \frac{F(z)D(z)r(z) + F(z)u_o r(z) + \delta_q(z)}{1 - P(z) + F(z)D(z)}. \qquad (B.10)$$

Using (3.23) again, the denominator of (B.10) reduces to

$$1 - P(z) + F(z)D(z) = [1 - P(z)][1 + D(z)G(z)]. \qquad (B.11)$$

Putting (B.11) into (B.10) and simplifying

$$\hat{y}(z) = \frac{D(z)G(z)}{1 + D(z)G(z)} r(z) + \frac{G(z)}{1 + D(z)G(z)} u_o r(z) + \frac{\delta_q(z)}{[1 - P(z)][1 + D(z)G(z)]}. \qquad (B.12)$$

Substituting (B.12) into (B.8) and simplifying

$$\begin{aligned}\delta(z) &= G(z)w(z) + P(z)G(z)u_o r(z) + P(z)D(z)G(z)r(z) \\ &\quad - P(z)\left[D(z)G(z)r(z) + G(z)u_o r(z) + \frac{\delta_q(z)}{1 - P(z)}\right] \\ &= G(z)w(z) - \frac{P(z)}{1 - P(z)}\delta_q(z) \qquad (B.13)\end{aligned}$$

which is the required Eq. (3.40). If the quantizer input is within the quantizer range, the quantization noise can be modelled as an additive noise bounded by (2.3) or (3.29). Hence

$$\delta_q(z) = \delta(z) + \epsilon(z) \qquad (B.14)$$

and when it is substituted into (B.13), Eq. (3.41) results:

$$\delta(z) = G(z)w(z) - \frac{P(z)}{1 - P(z)}[\delta(z) + \epsilon(z)]$$

$$= G(z)[1 - P(z)]w(z) - P(z)\epsilon(z)$$

$$= F(z)w(z) - P(z)\epsilon(z). \qquad \text{(B.15)}$$

APPENDIX C - Steady-State Optimal State Feedback Gain and Steady-State Kalman Filter Gain Design Data

LQG Weighting Factor, ρ	State Feedback Gain, K†		Closed-Loop Pole Location
	K_1	K_2	
1	3.9437132×10^{0}	3.4342070×10^{0}	$0.7766737 \pm j0.1723916$
10^{1}	7.8107538×10^{0}	6.1651287×10^{0}	$0.6192474 \pm j0.2597469$
10^{2}	1.4251803×10^{1}	9.9519167×10^{0}	$0.3748959 \pm j0.3294612$
10^{3}	2.2987488×10^{1}	1.4202253×10^{1}	$0.0641674 \pm j0.2980895$
10^{4}	3.1315598×10^{1}	1.7674103×10^{1}	$-0.0558185, \quad -0.3811369$

†$K=[K_1, K_2]$ for the second-order system used in the simulation.

Table C.1 Steady-State Optimal State Feedback Gain and Closed-Loop Pole Location for Plant A.

LQG Weighting Factor, ρ	State Feedback Gain, K		Closed-Loop Pole Location
	K_1	K_2	
10^{0}	4.6907520×10^{3}	3.2771826×10^{3}	$0.6486278 \pm j0.1658227$
10^{1}	6.9642529×10^{3}	4.4843164×10^{3}	$0.4731929 \pm j0.2760125$
10^{2}	1.0766401×10^{3}	6.2276270×10^{2}	$0.1936004 \pm j0.3176768$
10^{3}	1.5124360×10^{3}	7.9498633×10^{2}	$-0.1129392 \pm j0.1630775$
10^{4}	1.8350356×10^{3}	9.0940454×10^{2}	$-0.6561361, \quad -0.0104963$

Table C.2 Steady-State Optimal State Feedback Gain and Closed-Loop Pole Location for Plant B.

Quantizer Loading Factor	b	Solution to Eq.(3.52) R_*	Kalman Filter Gain, L‡		Closed-Loop Pole Location
			L_1	L_2	
3σ	3	4.0065957×10^{3}	9.5522362×10^{-1}	3.1938773×10^{-1}	$-0.1170428 \pm j0.1637539$
	4	2.4713613×10^{4}	9.8841703×10^{-1}	6.2164998×10^{-1}	$-0.0180079,\ -0.5815675$
4σ	3	1.6819702×10^{3}	9.2307711×10^{-1}	1.5936071×10^{-1}	$-0.0064183 \pm j0.2637389$
	4	1.2046879×10^{4}	9.7959179×10^{-1}	5.1253587×10^{-1}	$-0.0428316,\ -0.4308518$

‡L=[L_1 L_2] for the second-order system used in the simulation.

Table C.3 Steady-State Kalman Filter Gain and Closed-Loop Pole Location for Plant A.

Quantizer Loading Factor	b	Solution to Eq.(3.52) R_*	Kalman Filter Gain, L		Closed-Loop Pole Location
			L_1	L_2	
3σ	3	8.1946430×10^{4}	9.5522475×10^{-1}	2.7108598×10^{-1}	$-0.0885093 \pm j0.1922026$
	4	5.3110288×10^{7}	9.8841727×10^{-1}	5.9455532×10^{-1}	$-0.0210925,\ -0.5491179$
4σ	3	3.3000510×10^{4}	9.2307758×10^{-1}	1.0115683×10^{-1}	$0.0302213 \pm j0.2756910$
	4	2.5502192×10^{7}	9.7959208×10^{-1}	4.7744733×10^{-1}	$-0.0535309,\ -0.3810659$

Table C.4 Steady-State Kalman Filter Gain and Closed-Loop Pole Location for Plant B.

APPENDIX D - Quantizer Scaling Design Data

Quantizer Loading Factor	b	V_{FS} $(R_{W_{max}}=10^3)$	Δ_{max} $(R_{W_{max}}=10^3)$
3σ	3	6.9224982	1.7306246
	4	5.5745843	6.9682304×10^{-1}
4σ	3	1.0684187×10^{1}	2.6710467
	4	7.9844193	9.9805241×10^{-1}

Table D.1 Adaptive Quantizer Scaling for Plant A.

Quantizer Loading Factor	b	V_{FS} $(R_{W_{max}}=10^5)$	Δ_{max} $(R_{W_{max}}=10^5)$
3σ	3	1.5306842	3.8267105×10^{-1}
	4	1.2025172	1.5031464×10^{-1}
4σ	3	2.4120721	6.0301803×10^{-1}
	4	1.7353691	2.1692113×10^{-1}

Table D.2 Adaptive Quantizer Scaling for Plant B.

LQG Weighting Factor, ρ	V_{FS} $(R_{W_{max}}=10^3, r_{max}=10)$
1	3.8660934x10'
10	1.9878268x10'
10^2	1.1502698x10'
10^3	10
10^4	10

Table D.3 Quantizer Scaling of Fixed A/D for Plant A with 3σ Loading and b=4.